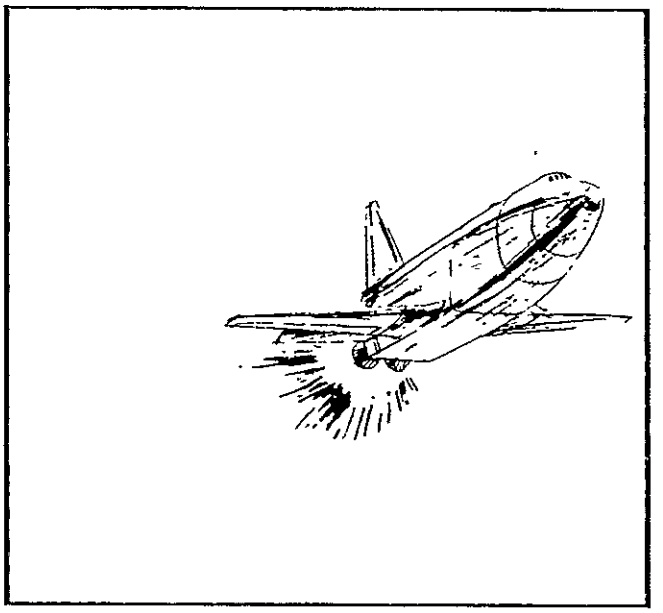
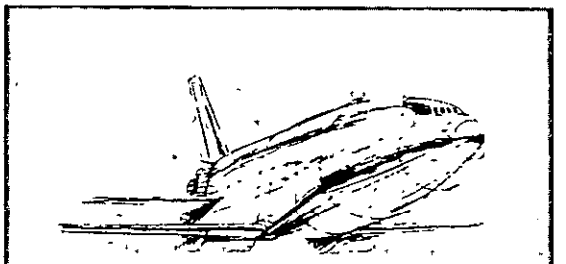


HONEYWELL

SYSTEMS & RESEARCH DIVISION



July 1971

CONTROLLER DESIGN TECHNOLOGY for the Space Shuttle Vehicle

Contract No. NAS8-25708

Honeywell Document 12238-IR1

12238-IR1

CONTROLLER DESIGN TECHNOLOGY
for the
SPACE SHUTTLE VEHICLE

July 1971

Prepared under Contract No. NAS8-25708 by

HONEYWELL INC.
Systems & Research Division
Minneapolis, Minnesota

for

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
George C. Marshall Space Flight Center

FOREWORD

This interim report, "Controller Design Technology for the Space Shuttle Vehicle; summarizes the study performed during the period 10 June 1970 to 1 June 1971 for the National Aeronautics and Space Administration, George C. Marshall Space Flight Center, under Contract NAS8-25708.

Dr. S. W. Winder of the Dynamics and Control Division of the Aero-Astrodynamics Laboratory was the technical monitor. The study was performed in the Systems and Research Division of Honeywell Inc. Dr. G. B. Skelton served as program manager. Dr. A. J. VanDierendonck was the initial principal investigator. He completed the study of iteration methods for the time-varying problem. Dr. C. A. Harvey, succeeded Dr. VanDierendonck as principal investigator, completed the work on controller simplification and conducted the study concerned with parameter variations. Dr. G. Stein became a coprincipal investigator and conducted the sensor choice study. Dr. Y. S. Lee contributed to the investigation of sensitivity to parameter variations. Mr. M. D. Ward assisted with computer programming and numerical analysis.

TABLE OF CONTENTS

TABLE OF CONTENTS (CONCLUDED)

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION I
INTRODUCTION AND SUMMARY

The goal of the research program summarized in this report was to develop a technology for control system design. The technology was aimed at providing control of flexure and rigid-body degrees-of-freedom of the Space Shuttle Vehicle within constraints of practicality. The constraints apply to the controller configuration and to the design methods as well.

At the time the program was initiated, it was anticipated that the Space Shuttle Vehicle would exhibit flexure control problems caused by large sizes of booster and orbiter sections, coupled modes of the vehicles in mated ascent, sensitivity to gust and maneuver excitation caused by large aerodynamic surfaces, and possible orbiter structural requirements imposed by reentry heating. The large aerodynamic surfaces could also cause a significant load relief problem at maximum dynamic pressure and larger drift dispersions than were present with Saturn Vehicles. Honeywell had achieved considerable success in treating such problems for large launch vehicles and large flexible aircraft. A stochastic constrained-response theory was developed, and its applicability to the control of a rigid booster was demonstrated in 1965 and 1966 (Ref. 1). The B-52 LAMS (Load Alleviation and Mode Stabilization) system was designed and flight-tested in 1966 and 1967 (Ref. 2). In 1967 and 1968, applicability of the method to a flexible launch vehicle was demonstrated (Ref. 3). In spite of the success achieved, the technology developed by 1968 was inadequate with respect to three control design problems: (1) controller simplification, (2) sensitivity to model inaccuracy, and (3) sensor complement choice.

Thus the specific goals of this study were to improve the design technology by providing:

- Practical controller simplification algorithms
- A mathematical method for implicitly including parameter variation constraints within quadratic optimization formulations
- A rigorous mathematical basis for understanding best sensor choice and location

The first objective is motivated by the difficulty of past efforts to simplify controllers. Simplification of the optimal controller in the LAMS program required a nine-man/month simulation program, and the optimal controller for the flexible launch vehicle was reduced by similar expensive trial and error methods. At that time, necessary conditions for optimizing constrained controller configurations were known (Ref. 4). These conditions provided a two-point boundary-value problem and computational algorithms for its solution. However, it was too expensive to use these algorithms for such problems as the flexible launch vehicle control problem, especially if several measurement complements or sets of parameter values were to be considered. Another drawback to these algorithms was the possibility of convergence to an arbitrary local minimum. Improved computational procedures for the solution of optimal control problems and simplification of optimal controllers had been developed at Honeywell by 1970 which held promise for treating the control problems of highly flexible vehicles. During the present study, these procedures were used as a basis for developing practical controller simplification algorithms. For the Space Shuttle Vehicle these algorithms may be used for controller design for mission phases such as the highly flexible orbiter during re-entry and cruise or for control during the mated ascent phase.

Mathematical models of two test vehicles were used for assessing the capability of the simplification procedures and improving them where necessary. A rigid representation of the Saturn V with a Voyager payload was used for a time-varying ascent study, and a model of the longitudinal axis of the C-5A with six flexure modes at a single flight condition was used for study of flexible-vehicle control. These choices were made to reduce the development cost, but the results of the two studies could be combined for actual application to Space Shuttle Vehicle controller design.

Controller simplification in both cases started with the solution of the stochastic constrained-response optimization problem with complete state measurement capability. For the launch vehicle, this solution was a set of time-varying gains and a time-varying deterministic input which defined the controller. This controller minimized a quadratic performance functional and, via quadratic equivalence, minimized as well a nonquadratic performance functional called the cost functional. This cost functional represents an upper bound on the likelihood of mission failure. A gradient iteration technique for controller simplification along with methods for choosing initial conditions for the iteration technique were developed and tested. These techniques proved to be quite capable for the example treated. The original controller utilizing ten time-varying feedbacks was simplified to a controller with five time-varying feedbacks.

Attempts at controller simplification for the flexible C-5A vehicle with an Implicit Function Method were unsuccessful. The cause of the failure was attributed either to lack of damping in the algorithm used or to severe sensitivity of performance to gain changes. Damping was added to the algorithm. This yielded some improvement, but not enough to provide a solution within assumed practicality constraints. Extrapolation based on the initial step of the Implicit Function Method was then used with gradient correction to achieve a satisfactory solution. A third method was also used successfully. This method, called the Incremental Gradient Method,

incorporates desirable features of the Implicit Function Method and usual gradient techniques.

The description of both phases of controller simplification is presented in Section II and Appendices A through C.

Another area in which the design technology needed upgrading to be applicable to the Space Shuttle Vehicle was sensitivity to model inaccuracy. In every iteration of controller design there is a degree of uncertainty in the data. Also, certain dynamics such as high order flexure and fuel-sloshing modes are generally ignored to make the design problem tractable. Thus, practical controller design must recognize and be tolerant of model inaccuracies. In this study such inaccuracies were assumed to take the form of unknown parameters. Two directions were pursued to yield formulations of the quadratic optimal control problems which would implicitly include parameter uncertainty constraints. One direction was to determine properties of the optimal performance surface over a segment of parameter space which could be used to derive optimal insensitive controllers. An optimal insensitive controller is one that is optimal for a given value of the parameters defining the system and minimizes the maximum of the performance index over all admissible values of the parameters. A necessary condition for an optimal insensitive controller was derived. This condition was shown to be locally sufficient. The nature of these conditions led to the development of an algorithm for computing approximately optimal insensitive controllers. The utility of this computational approach was tested on the C-5A example, and significant reduction in sensitivity was achieved.

The other direction of the parameter variation study was an investigation of the effectiveness of compensators in reducing sensitivity. This study led to choosing control parameters to match the performance of dominant dynamics of the compensated system to the performance of the optimal uncompensated system over the range of system parameters. This approach is intuitively

appealing. But for high-dimensional systems the approach is of questionable value.

Analyses performed in these methods of including parameter variation constraints are described in Section III and Appendix D. Results derived for approximately optimal insensitive controllers for the C-5A example are presented in Appendix E.

A third area of technology improvement concerns the choice of types and locations of sensors to generate feedback signals for practical control systems. This so-called "sensor-choice problem" arises because it is economically prohibitive in most applications to measure all system states, particularly the rates and displacements of flexure modes. The problem is unsolved because it is theoretically and computationally difficult to determine basic performance capabilities of a set of sensors. To do so requires the solution of two coupled optimization problems — (1) optimal location of the instruments and (2) optimal design of a practical controller to utilize the instruments. Both are complex problems when a quadratic performance functional is used as the measure of quality. Available methods are based on controller optimization routines such as those discussed in Section II nested within iterative search procedures for best sensor locations (Ref. 5, 11).

In this study, the sensor choice problem was approached from the viewpoint of finding alternate quality measures which would be more convenient computationally yet still provide meaningful indications of performance capability. Two measures were proposed, both based on the pole-placement capability of the sensor complement. The first measure is the maximum number of closed-loop poles, p_{max}, which can be placed arbitrarily, and the second is a measure of deviation of the remaining nonarbitrary poles from specified desirable locations. General equations were derived for these quantities, and a computational pole placement algorithm was developed for their solution. Computational feasibility of the measures was established. However, their

practical utility as quality measures, particularly their correlation with quadratic cost, remain to be verified.

Analyses performed on the sensor choice problem are described in Section IV and Appendices F, G and H.

SECTION II
CONTROLLER SIMPLIFICATION

The aim of this phase of the study was to develop economically feasible methods for computing simplified controllers for the Space Shuttle Vehicle. The simplified controllers were to be derived from optimal controllers. Simplification was defined to be the reduction in the output measurements required and, in the case of time-varying gains, reduction in the complexity of the time-variations to specified parametric representations. Of course, the simplified controllers were to maintain as much as possible the desired performance of the optimal controllers.

BACKGROUND

The mission of the Space Shuttle Vehicle is such that at times the control problems are similar to those of a large flexible launch vehicle and at other times to those of large flexible aircraft. The stochastic constrained-response formulations of these two types of control problems had been derived in References 1 and 2. Resulting optimal controllers had been shown to provide very desirable performance. The measures of performance and the mathematical models for large flexible launch vehicles and large flexible aircraft have several common features. The major distinction from a mathematical viewpoint is that the control problem for the launch vehicle is a finite-time problem with significantly time-varying dynamics, while the aircraft can be considered to fly at a single flight condition for a long enough period of time that the dynamics may be represented by a constant-coefficient model, and a steady-state performance functional is of interest. Common features include linear dynamics, many degrees of freedom, stochastic disturbance models, and physically meaningful performance functionals

which are generally not expressable as integrals of quadratic functions of (linear) responses. These common features suggest that, in theory, simplification methods which are successful for one problem will also be successful for the other. In practice, the computational costs for the two problems differ significantly. For example, with a quadratic performance functional, the cost of computing an optimal launch vehicle controller for a 23rd-order Saturn model is approximately 50 times the cost of computing an optimal controller for a 23rd-order model of a flexible aircraft at a single flight condition. These facts motivated the separation of the controller simplification study into two parts, one part dealing with the time-varying aspect of the problem and the other dealing with the high dimensionality of the mathematical models associated with flexure.

In addition to the difference in computation costs between the time-varying and constant-coefficient problems, there is a difference in computer storage requirements. For problems with the same number of degrees of freedom, the time-varying problem requires much more computer storage than does the constant-coefficient problem. Computation time considerations led to the choice of a rigid model of a launch vehicle as a test vehicle for the time-varying aspects of this study. This permitted savings in computer costs because of the reduction in order of the system from the model with flexure and also because larger sampling intervals could be used. The flexure degrees of freedom were retained in the constant-coefficient model.

Computation time and storage requirements also influenced the type of techniques chosen for simplification in the two problems. The high cost of computation for the time-varying problem limited the acceptable methods to first-order, that is, methods which utilize at most first-order derivatives. Large storage requirements impose the same limit for the constant-coefficient problem if the number of nonzero parameters (gains) permitted in the simplified configuration is large. If the number of such parameters is not large, then second-order methods are candidates for consideration.

SIMPLIFICATION PROCEDURES

The starting point for the procedures is a solution to the following optimization problem. Dynamics of the system are represented by the linear differential equation

$$\dot{x} = Fx + G_1 u + G_2 \eta \tag{1}$$

where x is the state vector, u is the control vector and η is a white-noise vector of disturbances with known means and covariance. Responses to be controlled are linear combinations of the states and controls given by

$$r = Hx + Du \tag{2}$$

The performance index is a functional of the form

$$J = f_1[R(T), S(T)] + \int_0^T f_2[R(t), S(t)]dt \tag{3}$$

where

$$R(t) = \bar{r}(t)[\bar{r}(t)]^T, \quad \bar{r}(t) = E\{r(t)\} \tag{4}$$

$$S(t) = E\{[r(t)-\bar{r}(t)][r(t)-\bar{r}(t)]^T\} \tag{5}$$

The superscript T, denotes the transpose. The optimization problem is to choose u to minimize J. For the ascent problem, the time T in (3) is finite and F, G_1, G_2, H, D, the function f_2, and the mean and covariance of the noise may be explicit functions of time. For the flexible-aircraft problem, the coefficient matrices, the function f_2, and the noise characteristics are constant, f_1 is zero and T in the integral in (3) is infinite.

The "quadratic problem" occurs when f_1 and f_2 in (3) are linear functions of R and S, since these matrices are quadratic in the responses. The solution of the quadratic problem is

$$u = K^*x + f \tag{6}$$

where K* and f satisfy certain determining equations. For simplicity of discussion, let us consider the constant-coefficient problem with $f_2 = \mathrm{Tr}[QS]$ where Tr denotes the trace. Also, for simplicity assume $E\{\eta\} = 0$ which implies $f = 0$. This last assumption does not represent a restriction on the simplification problem. Inclusion of a deterministic input in the controller is not a real complication. The determining equations for K are

$$0 = D^T Q(H + DK^*) + G_1{}^T P \tag{7}$$

and

$$0 = (F + GK^*)^T P + P(F + GK^*) + (H + DK^*)^T Q(H + DK^*) \tag{8}$$

Analytically, controller simplification consists of replacing the matrix K* with a matrix K of a constrained form which has less independent parameters than the number of elements in K*. The resulting controller, of course, should maintain as low a value of J as possible. Thus at this point, J may be considered as a function of K to be minimized subject to the constraining configuration imposed on K. The procedures used in this study for performing this task may be classified as gradient iteration and parametric techniques.

Gradient Iteration Technique

The gradient iteration technique consists of applying gradient methods to the minimization with respect to K of the Hamiltonian

$$H(X, P, K) = Tr\{Q(H + DK)X(H + DK)^T + P[(F + G_1K)X + X(F + G_1K)^T + N]\}\dagger \tag{9}$$

where X denotes the state covariance matrix and $N = E\{G_2 m^T G_2^T\}$.

The necessary conditions for optimality of K derived by Axsäter (Ref. 4) are

$$\dot{X} = \frac{\partial H}{\partial P}, \qquad \dot{P} = -\frac{\partial H}{\partial X}, \qquad \frac{\partial H}{\partial K} = 0 \tag{10}$$

with K constrained to be of the desired configuration. If complete measurement of the state is permitted, the first of equations (10) is uncoupled from the last two. With complete measurement of the state not permitted, the three equations are coupled. Thus at each iteration, the first two equations are integrated to find X and P. Then the gradient $\partial H/\partial K$ is evaluated. When this gradient is nonzero, a step is taken to reduce the value of the Hamiltonian.

Existence of several local minima for the constrained minimization problem is indeed possible. But if an initial value for K is known which is sufficiently close to the global minimum, the gradient methods have the desired property of convergence to the global minimum with sufficiently small steps. Thus, the successful application of this technique depends on having a good initial value for K, proper choice of step size, and, of course, the capability to compute gradients economically.

Two methods were used to obtain initial values of K for the iteration studies. The assumed configuration for the controller was u = Kz where

$$z = Mx \tag{11}$$

---

†The second term in the trace is identically zero for the steady-state problem. It is included here to indicate the presence of its analog in the time-varying problem.

represented a vector of measurements. The first method was to choose K to be the projection of K* on M

$$K = K^*M^T(MM^T)^{-1} \tag{12}$$

When the components of z are components of x, the choice in (12) merely deletes the gains in K* which multiply states that are not measured. The second method was based on approximating a Kalman Estimator used in conjunction with K* by the gain K.

Essentially, this method involves finding a good match between the open-loop responses of Figures 1(a) and 1(b), for a very small measurement noise ξ. Figure 1(a) yields a response in the frequency domain of

$$U(j\omega) = Y(j\omega)Z(j\omega) \tag{13}$$

where Y(s) is the Laplace transform of the system shown. Specifically

$$Y(s) = K(sI - F - G_1K^* + LM)^{-1}L \tag{14}$$

where L is the Kalman gain matrix.

Figure 1(b) yields a response of

$$U(j\omega) = KZ(j\omega) \tag{15}$$

where K is not a function of frequency.

Two methods to match these responses were tried. The first was to set the static gains equal to the d-c gains of the dynamic system. That is

$$K = Y(0) \tag{16}$$

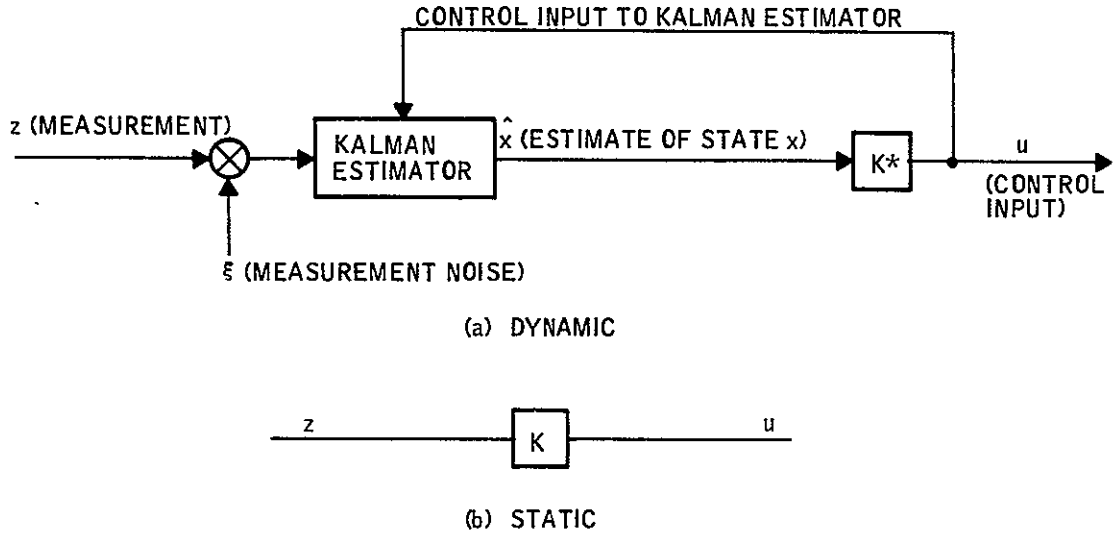


Figure 1. Dynamic and Static Gain Systems

The second was to compute a weighted average of $Y(j\omega)$ over all frequencies. That is

$$K = \frac{\mathrm{Re} \int_0^\infty Y(j\omega) \left| Z(j\omega) \right| d\omega}{\int_0^\infty \left| Z(j\omega) \right| d\omega} \tag{17}$$

where Re denotes "the real part of." These methods were actually applied to the time-varying example discussed below.

The choice of step size is a problem common to all gradient iteration methods. Furthermore, it is well-known that the path of steepest descent is dependent on the coordinate system chosen. For simplicity of exposition, consider K to be a vector with components k_i. The gradient iteration may be expressed as

$$K(p+1) = K(p) - \epsilon \nabla_K H \Big|_{K=K(p)}, \qquad \epsilon > 0 \tag{18}$$

where p denotes the stage of the iteration, and ϵ denotes the step size. A sketch of possible behavior of such an iteration method is shown in Figure 2 for a two-dimensional K. In this sketch, points A, B, C, D and E indicate possible values of K(p+1) corresponding to different choices of ϵ. If the value of ϵ chosen is less than or equal to the value corresponding to C, the gradient at K(p+1) would be directed in the general direction of the point where H = 0. A common manner of choosing ϵ is to perform the minimization of H[K(p+1)] with respect to the parameter ϵ. The result for K(p+1) on the segment from K(p) to C is depicted as point B. For problems such as the time-varying control problem, even this one dimensional minimization can be costly so that an approximation is generally accepted. It is also clear from Figure 2 that admissible directions of the step from K(p) to yield decreases in H are all directions with positive projections on the vector from K(p) to B. In the example treated in this study such a modification was introduced. Equation (18) was replaced by

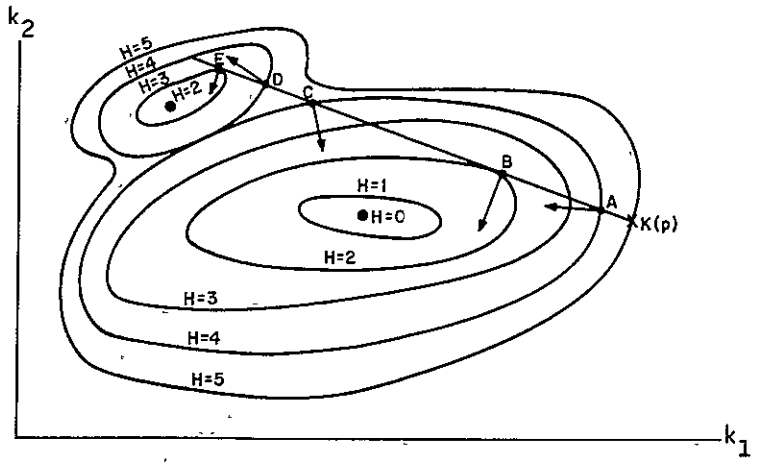


Figure 2. Gradient Step Possibilities

$$K(p+1) = K(p) - \epsilon(\text{diag}\,|k_i(p)|)\left[\nabla_K H[\,|\nabla_K H|\,]^{-1}\right]_{K=K(p)}, \quad \epsilon > 0 \tag{19}$$

where diag $|k_i(p)|$ denotes the diagonal matrix with $|k_i(p)|$ as its i^{th} diagonal element. In this form ϵ represents the maximum percentage change in any element of K(p). This iteration equation proved to be successful in the example.

The equations for computation of the gradient in the example are described in the discussion of the example. The required computation is the solution of a covariance and a co-state equation. For the steady-state problem, the gradient of H can be computed as follows. The steady-state covariance matrix, X, satisfies

$$(F + G_1K)X + X(F + G_1K)^T + N = 0 \tag{20}$$

Thus, $H = \text{Tr}(H+DK)^T Q(H + DK)X = J$. The adjoint matrix A corresponding to X satisfies

$$(F + G_1K)^T A + A(F + G_1K) + (H + DK)^T Q(H + DK) = 0 \tag{21}$$

Then

$$\frac{\partial J}{\partial K_{ij}} = \text{Tr}\left[2\,(H + DK)^T\, Q\, DE^{ij}\, X + (H + DK)^T\, Q\, (H + DK)\,\frac{\partial X}{\partial K_{ij}}\right] \tag{22}$$

where $E^{ij} \triangleq \dfrac{\partial K}{\partial K_{ij}}$ and $\dfrac{\partial X}{\partial K_{ij}}$ is defined by

$$(F + G_1K)\frac{\partial \dot{X}}{\partial K_{ij}} + \frac{\partial \dot{X}}{\partial K_{ij}}(F + G_1K)^T + G_1E^{ij}\,X + \dot{X}\,(G_1E^{ij})^T = 0. \qquad (23)$$

Using the adjoint property of X and A

$$\frac{\partial J}{\partial K_{ij}} = \mathrm{Tr}\,[2\,(H + DK)^T\,Q\,DE^{ij}\,X + A\,(G_1E^{ij}\,\dot{X} + X\,(G_1E^{ij})^T]$$

$$= 2\,\mathrm{Tr}\,\{[(H + DK)^T\,QD + AG_1]\,E^{ij}\,X\} \qquad (24)$$

Thus, to evaluate all of the first partials of J, we need only evaluate the two covariance-type equations (20) and (21) and the algebraic equations (24). This convenient property was first discovered by T. L. Johnson (Ref. 5).

Parametric Techniques

In this technique developed by G. Stein (Ref. 6), a general gain matrix K is partitioned into two orthogonal components:

$$K = (K^1 + K^3)M + \lambda K^2 \qquad (25)$$

where $(K^1 + K^3)M(K^2)^T = 0$.

The first component $(K^1 + K^3)M$ satisfies the constraint and hence represents gains to be retained. The second component λK^2 consists of gains to be discarded. The scalar parameter, λ, is introduced in equation (25) to permit discarding the second component while maintaining one constraint on the first component. For generality, the first component is divided into two orthogonal components K^1 and K^3 where K^1 represents gains to be optimized and K^3 is fixed. The necessary condition for optimality of K^1 with respect to the constrained problem is

$$\left.\frac{\partial J[(K^1 + K^3)M + \lambda K^2]}{\partial K^1}\right|_{\lambda=0} = 0 \tag{26}$$

Solutions to this equation may be obtained as noted by D. K. Scharmack (Ref. 7) by starting with a K in (25) with $\lambda = 1$ that satisfies

$$\frac{\partial J(K)}{\partial K^1} = 0 \tag{27}$$

and choosing K^1 as a function of λ such that equation (27) is maintained for $0 \leq \lambda \leq 1$. The most appealing choice for $\lambda = 1$ is the K that minimizes J globally; namely

$$K^* = (K^1(1) + K^3)M + K^2 \tag{28}$$

According to the Implicit Function Theorem (Ref. 8), $K^1(\lambda)$ satisfying (27) is defined by

$$\frac{dK^1(\lambda)}{d\lambda} = -\left[\frac{\partial^2 J[(K^1 + K^3)M + \lambda K^2]}{\partial K^1 \partial K^{1T}}\right]^{-1} \frac{\partial^2 J[K^1 + K^3)M + \lambda K^2]}{\partial K^1 \partial \lambda} \tag{29}$$†

Some computational difficulties were experienced in applying this technique using numerical integration of (29) for the C-5A example. It was noted that equation (29) is equivalent to

†Equations (26), (27) and (29) make sense in vector matrix notation if K^1 is written as a column vector, which is assumed.

$$\frac{d}{d\lambda}(\nabla J) = 0 \tag{30}$$

where ∇J is abbreviated notation for the partial of J with respect to the gains in K^1. As a differential equation in λ, equation (30) is neutrally stable, so a damping term of $a(\lambda)$ J was added to the right-hand side. The scalar $a(\lambda)$ was chosen to be positive since the integration is in the negative λ direction. This modification gave some improvement, but not complete success. A hybrid technique using this method of finding $K^1(.95)$ from $K^1(1)$, then extrapolating to $K^1(0)$ and then using gradient correction was then used successfully.

A variation on this theme was developed by A. J. Van Dierendonck (Ref. 9). This variation is called the incremental gradient algorithm. This algorithm begins with the same values of K and K^1 at $\lambda = 1$. Then λ is incremented to $\lambda + \Delta\lambda$ (with $-1 < \Delta\lambda < 0$). Several gradient steps are then taken to obtain $K^1(\lambda + \Delta\lambda)$. Another increment in λ is then taken with linear prediction used for K^1. Again gradient corrections in K^1 are made and the process continued until $\lambda = 0$. This method was also tried successfully on the C-5A example.

A comparison of the three parametric methods is illustrated for a simple example in Figure 3. For ease of representation, it is assumed K is a two-vector and $K^1 = k_1$, $K^2 = k_2$. In realistic examples, each of these axes may be multidimensional subspaces. The sketch is completely hypothetical. The Implicit Function Solution shown is assumed to be exact. The other two paths are intended to be only indicative of the actual situations.

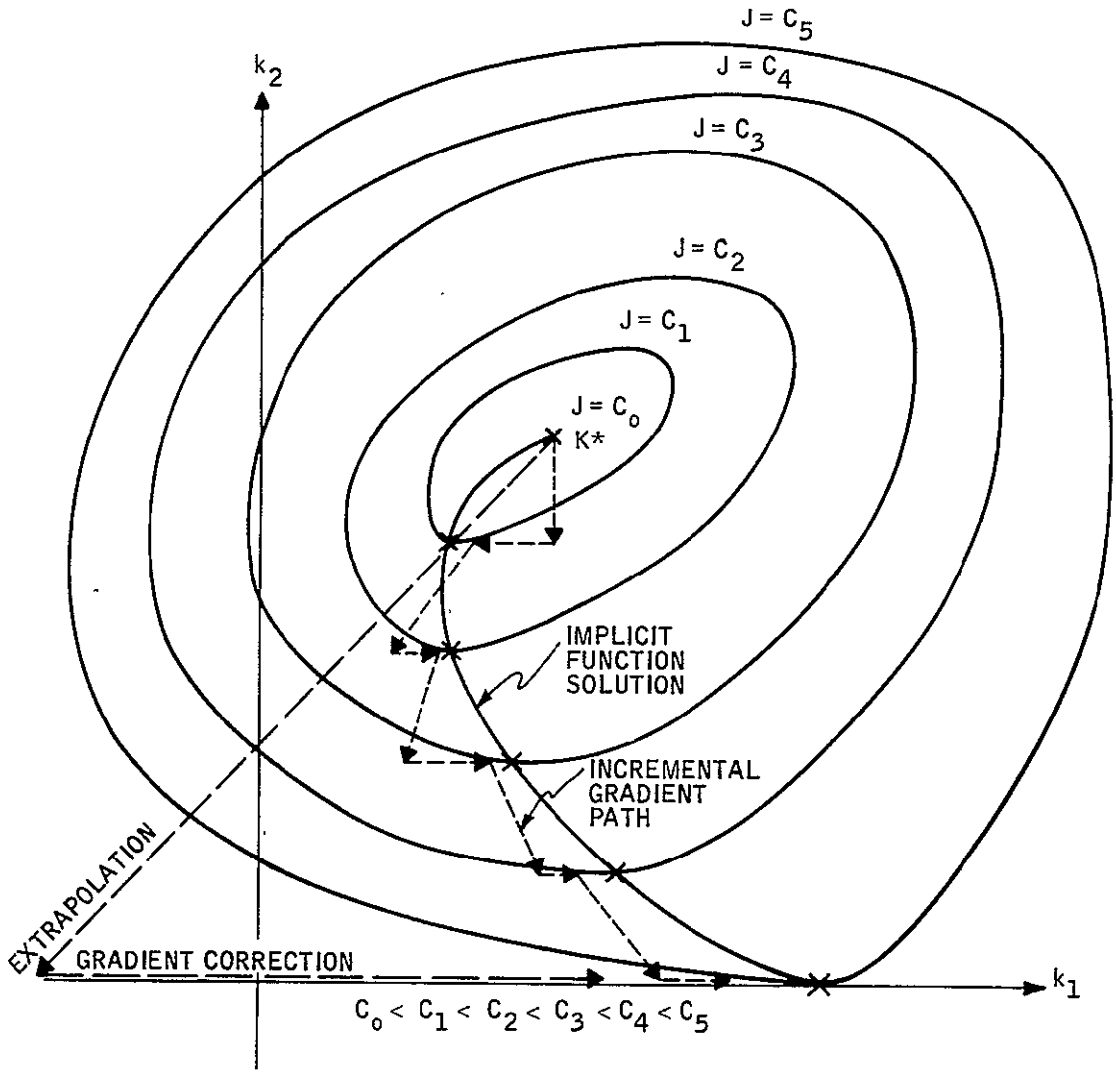


Figure 3. Geometric Interpretation of Paràmetric Techniques

SIMPLIFICATION AND QUADRATIC EQUIVALENCE

In the study of the time-varying example, it was discovered that the quadratic approximation to a nonquadratic performance functional used in quadratic equivalence is dependent on the constraints imposed. The following example demonstrates this phenomenon.

Consider minimizing the function $f(x, y) = x+y+3y^2$ for (x, y) belonging to certain sets, say K_1 and K_2. We may think of these sets as representing sets of attainability for a dynamical system and suppose that K_1 and K_2 correspond to complete and incomplete measurements respectively. Then since measurements could be ignored in the complete measurement case, K_2 would be contained in K_1. In fact, let us assume K_2 is a proper subset of K_1. To give this problem the interpretation of a stochastic optimal control problem, we may suppose that x and y represent covariances of two responses. This would imply $x \geq 0$ and $y \geq 0$. Now suppose that

$$K_1 = \{(x, y) : x \geq 0 \text{ and } xy \geq 1/3\}$$

$$K_2 = \{(x, y) : x \geq 0 \text{ and } xy \geq 1\}$$

The minimum of $f(x, y)$ for (x, y) in K_1 occurs at $x = 1$, $y = 1/3$. This point also yields the minimum for the "quadratic" function $f(x, y; q_1, q_2) = q_1x + q_2y$ with $q_1 = 1$, $q_2 = 3$. Note that the vector $(q_1, q_2) = (1, 3)$ is a normal to the boundary of K_1 at $(x, y) = (1, 1/3)$. To interpret $f(x, y; q_1, q_2)$ as a "quadratic" cost function, we consider x and y as variances of responses V_1 and V_2 so that $f(x, y; q_1, q_2)$ is quadratic in the responses. For (x, y) belonging to K_2 the minimum of $f(x, y)$ occurs at $x = 2$, $y = 1/2$ and normal at that point is $(1, 4)$. Thus the equivalent quadratic for K_1 is $f(x, y; 1, 3)$ whereas for K_2 it is $f(x, y; 1, 4)$. Figure 4 shows the relevant geometry for this example.

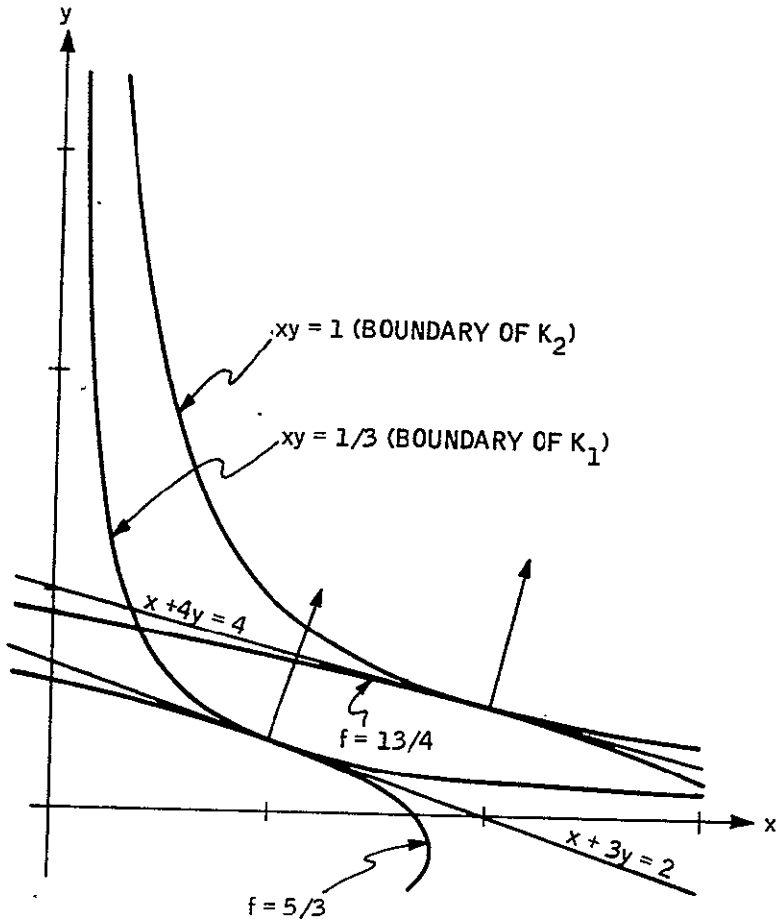


Figure 4. Constraints and Quadratic Equivalence

This phenomenon may be quite significant for problems in which quadratic equivalence and simplification are considered. The simplification algorithms for such problems should probably utilize the gradients of the nonquadratic performance functional, at least in the final stages and not rely on the quadratic approximation throughout the simplification procedure. This was clearly the case in the study of the following example.

TIME-VARYING EXAMPLE

The vehicle used in this example is the rigid-body representation of the vehicle described in Reference 3. The resulting model is 10th order and includes rigid body, wind filter, distributed wind loads, and gimbal angle. For this model of the vehicle, a sampling rate of 10 samples per second was adequate rather than the 50 samples per second required for the original model which included three flexure modes.

Initial optimization runs were made assuming complete measurement capability (i.e., every vehicle and wind state can be measured). Using the concept of quadratic equivalence, three iterations yielded a controller whose upper bound of the probability that mission failure (as described in Reference 3. would occur was 1.7×10^{-4}. The upper bound on the probability of mission failure is denoted by J*, while J** denotes a quadratic cost functional. It was assumed that ϕ (pitch), $\dot{\phi}$ (pitch rate), z (drift), $\dot{z}$ (drift rate), and β (gimbal angle) could be measured directly but that the measurements were noisy. A Kalman Estimator was derived to extimate ω and x (wind states) and x_1, x_2 and x_3 (load-distribution states). The costs, J* and J** were computed with the optimum gains and the Kalman Estimator. Although the quadratic cost increased slightly over the perfect-sensing case, the upper bound on the probability of mission failure lowered significantly from 0.7×10^{-4} to 0.3×10^{-6}.

The desired form of simplified controller was chosen to be feedback of the measured quantities without using compensators. The simplest choice of initial gains for the gradient iteration method is the perfect sensing gains for states which can be measured and zero for the states not measured. For this example, this choice produced results comparable to the perfect-sensing gains. This was understandable since, for this rigid-body model, essentially all costs were due to the second bending moment at maximum dynamic pressure, and this can be controlled with pitch, pitch rate, and the gimbal angle. These results also explain why the Kalman Estimator did not appreciably change the results from those for perfect sensing.

To create a problem to test the gradient iteration method, the terminal-drift constraint was reduced by a factor of three from 30,000 to 10,000 meters. It was found that this new constraint was easily met. Hence, this did not create a meaningful problem.

Thus, another approach was taken. Instead of tightening the constraints, the magnitude of the disturbance input was increased by doubling the standard deviation of the wind. After changing the quadratic weights, a new optimal controller was computed that met the original constraints. This controller resulted in an upper bound on the probability of mission failure (J*) of 0.4299×10^{-33}. The derivatives of this probability with respect to the individual responses indicated that quadratic equivalence was very nearly achieved.

The value of J* for the controller with perfect-sensing gains for measured states and zero gains for unmeasured states was 0.7494×10^{-6}. The quadratic cost (J**) for this set of gains was 0.9344×10^{8} compared to 0.8664×10^{8} for perfect sensing. The differences in these costs are sufficient to yield a meaningful test of the gradient-search method even though the values of J* are small.

Detailed equations for the gradient iterations are given in Appendix A. One step of the search based on minimizing the quadratic cost J** lowered this cost to 0.9265×10^{8}. The J* cost, however, went up to 0.2188×10^{-5}. This was because the quadratic Q(t) used for the perfect-sensing controller is not necessarily the best for the constrained system as indicated above.

The cost J* can be minimized directly in a gradient search as shown in Appendix A. This method proved very successful in reducing the J* cost from 0.7494×10^{-6} to 0.1266×10^{-8} to 0.1385×10^{-12} to 0.110×10^{-17} in three iterations.

Initial conditions for the gradient iterations were obtained by approximating the Kalman Estimator with time-varying gains. Both methods (DC and averaging over frequency) were used with data at 33 instants of time, 5 seconds apart. Linear interpolation was used between these points. The first method [K(t) = Y(0, t)] proved to be quite poor because the pitch and pitch rate states are of higher frequency. Thus these DC gains were too far off. The resulting quadratic cost was about four orders of magnitude higher than for perfect sensing. The probability of mission failure was certainty (J*=88). The second method proved to be somewhat better. The quadratic cost was up less than one order of magnitude. However, probability of mission failure was still certainty (J* = 2). It was still close enough for the gradient search. One gradient step was taken, and the J* cost was lowered to 0.354.

For problems where the unmeasured states are important, which was not the case here, this method of finding initial conditions appears to be satisfactory. Use of more data points in time and frequency would probably give better results. The weighted averages were computed using only 12 values of ω between 0 and 10 radians per second.

As a severe test, the gradient method was tried with the d-c gains as initial gains. Using appropriate normalization of the gradient vector and using the initial quadratic weights, we were able to reduce the J* cost to 0.373 in three steps. For the first two steps, the gradient was normalized for each gain over all time separately. This had the effect of changing the shape of each gain versus time. If this hadn't been done, and the gradient had been normalized over all the gains over all time, only the shape of one gain (drift) would be changed, since the gradient for that gain always dominated the others.

For the first two steps (J* = 88 to J* = 0.525) the original quadratic weights were used to drive the response covariances down to a reasonable level. The probability calculations were not valid for the original covariance values. Thus, minimizing the J* cost was unsatisfactory because the gradient was invalid. The quadratic cost in the first two steps was reduced four orders of magnitude.

After the first two steps, the J* cost was low enough for valid calculations of the partial derivatives of J* with respect to covariance responses so that they could be used for the quadratic weights. One more step reduced the J* cost to 0.373.

The gradient iterations were terminated with this set of gains since previous iteration had been successful in reducing the cost to a desirable value from such a value of J*.

Equations for simplification of the time-varying nature of the gains were derived and appear in Appendix B.

The time-varying gains obtained with gradient iterations from the three initial conditions chosen are shown in Appendix C. These figures demonstrate that the second and third iterations from the perfect measurement gains were

of a fine-tuning nature. The figures also demonstrate improved behavior starting from the averaged approximation rather than the DC approximation to the Kalman controller.

CONSTANT COEFFICIENT EXAMPLE

The test vehicle considered is a 23rd-order model of the C-5A pitch dynamics. The model consists of the short-period mode, six flexure modes, two second-order Kussner lift-growth modes, two wind states, and three first-order actuators. The three control surfaces are inboard elevators, ailerons and spoilers. The response vector consists of stresses and stress rates at five stations and normal accelerations at four stations. A more detailed description of the C-5A model is given in Appendix E. A possibly practical controller form was chosen on the basis of previous analysis of this model performed on the LAMS program (Ref. 10). This form is shown in Figure 5.

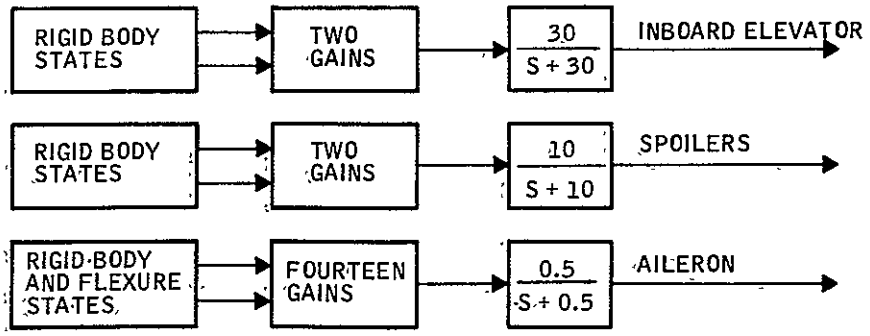


Figure 5. Controller Configuration for C-5A

Attempts to use the Implicit Function Method described above required small steps in λ ($\Delta\lambda = -0.05$). Investigation revealed that the gradient of J with respect to gains in K^1 (denoted by ∇J) was highly sensitive to changes in gains. Small percentage changes in elements of K^1 gave rise to orders-of-magnitude changes in components of ∇J. The matrix $\nabla^2 J$ was rather ill-conditioned with a range of 10^{10} for the magnitudes of its eigenvalues. The original algorithm with $\Delta\lambda = -0.05$ was unsuccessful at $\lambda = 0.80$. The predicted controller was unstable. The algorithm was modified to include damping, and two gains which appeared to oscillate in a diverging manner were held constant (put in K^3). With these modifications, satisfactory performance was obtained to $\lambda = 0.50$. The gains at $\lambda = 0.45$ were unsatisfactory. They yielded a non-definite matrix of second partials $\nabla^2 J$ and consequently unsatisfactory gains at $\lambda = 0.40$.

The Implicit Function Method was discarded in favor of extrapolation followed by gradient corrections. A predicted set of gains for K^1 at $\lambda = 0$ was computed from the equation

$$K^1(0) = 20\, K^1(.95) - 19\, K^1(1.0) \tag{31}$$

Ten gradient corrections were made from this predicted gain. The magnitude of the largest component of ∇J for the resulting gain was less than the magnitude of the largest component of ∇J at $\lambda = 1.0$. This was considered to be the accuracy that could be expected from the numerical calculations.

Finally, the incremental gradient method was used with a slightly different controller configuration. This configuration consisted of allowing flexure feedbacks to the elevator and spoilers. For this calculation $\Delta\lambda$ was chosen to be -0.10, and four conjugate gradient steps were used for correction at each value of λ. This method proved to be quite successful for the case studied.

Numerical results are shown in Tables 1 through 4. In Table 1, the behavior of the Implicit Function Method algorithm for the C-5A is displayed. Without damping, the algorithm failed at $\lambda = 0.8$. The modified algorithm to include damping was slightly better, but failed at $\lambda = 0.70$. A second attempt with the modified algorithm was made in which two gains corresponding to flexure-mode displacements were constrained to be constant. Greater success was achieved in that failure did not occur until $\lambda = 0.35$. However, the last step was very poor, and simple extrapolation proved to be much better. This motivated the extrapolation to $\lambda = 0$ followed by successive gradient steps. The results appear in Table 2, where step number zero is the result of extrapolation from $\lambda = 0.95$ to $\lambda = 0$. The gradient steps were terminated at the point where the magnitude of the gradient vector, as measured by its largest element, became less than the magnitude of the gradient at $\lambda = 1$. The gains for this controller are given for comparison with the original gains and those from extrapolation in Table 3. Finally in Table 4, the performance of the Incremental Gradient Algorithm is listed. Flexure feedback to all control inputs was permitted in this case. The cost in the second column is that obtained before gradient correction while column three indicates the cost after gradient correction.

The magnitudes of costs and gradients listed in Tables 1 through 4 may appear somewhat large considering the fact that the gradient for the optimal control should be zero. However, these magnitudes are arbitrary in the sense that the scaling of J and hence of ∇J is arbitrary. To obtain a calibration of J with respect to physical considerations, the value of J corresponding to no stress and stress rate costs and standard deviations of 0.1 g for the accelerations at the four stations is 8500. Significance of the magnitude of ∇J computed for the optimal controller with complete measurement feedback may be deduced as follows. Suppose that K* denotes the optimal gain matrix which will be considered as a gain vector. In a sufficiently small neighborhood of K* the cost may be represented as

$$J(K) = J(K^*) + 1/2\,(K-K^*)\cdot \nabla^2 J\big|_{K^*}(K-K^*)$$

Then $\nabla J|_K = \nabla^2 J|_{K^*}(K-K^*)$. Now suppose $K-K^* = EK^*$

where E is a diagonal matrix. The diagonal elements of E may be considered as relative errors in the components of K* and should be small if K approximates K*. For the computed optimal controller each element of E was less than 0.019 in magnitude indicating an accuracy of approximately 2% or better for each gain.

Table 1. Behavior of Implicit Function Method Algorithm for the C-5A

Original Algorithm			Modified Algorithm			
			Unconstrained K's		Constrained K's	
λ	J	$\max\|(\nabla J)_i\|$	J	$\max\|(\nabla J)_i\|$	J	$\max\|(\nabla J)_i\|$
1.00	14377	$.28 \cdot 10^5$	14377	$.28 \cdot 10^5$	14377	$.28 \cdot 10^5$
.95	14429	$.88 \cdot 10^5$	14429	$.43 \cdot 10^5$	14427	$.14 \cdot 10^5$
.90	14595	$.98 \cdot 10^5$	14590	$.45 \cdot 10^5$	14582	$.41 \cdot 10^5$
.85	14884	$.42 \cdot 10^6$	14862	$.91 \cdot 10^5$	14848	$.28 \cdot 10^5$
.80			15614	$.23 \cdot 10^6$	15233	$.35 \cdot 10^5$
.75			16092	$.34 \cdot 10^6$	15749	$.12 \cdot 10^5$
.70					16416	$.31 \cdot 10^5$
.65					17248	$.57 \cdot 10^5$
.60					18275	$.53 \cdot 10^5$
.55					19470	$.52 \cdot 10^5$
.50					21170	$.18 \cdot 10^6$
.45					24366	$.43 \cdot 10^6$
.40					86984	$.95 \cdot 10^7$
.40[a]					26482	$.53 \cdot 10^6$

[a]Indicates extrapolated set of gains

Table 2. Extrapolation plus Gradient Algorithm

Step Number	J	$10^{-5} \cdot \max \lvert(\nabla J)_i\rvert$
0	97125	51
1	94789	34
2	93389	18
3	92373	22
4	88926	13
5	88644	10
6	88445	6.3
7	83880	0.5
8	72191	3.8
9	70685	0.9
10	70679	0.275

Table 3. Gain Comparison

Gains	Controller		
	Optimal with Complete Feedback	Extrapolated (step 0)	Final (step 10)
$K_{1,1}$	-.02983	-.00780	-.00545
$K_{1,2}$	-.01574	.0103	.0114
$K_{1,3}$	-.2388	.00520	.00899
$K_{1,4}$	.06796	.0900	.0842
$K_{1,5}$	2.837	-1.06	-1.06
$K_{1,6}$	.3137	.048	.100
$K_{1,7}$	-1.396	-4.90	-4.90
$K_{1,8}$	-.4093	-.175	-.023

Table 3. Gain Comparison (Continued)

Gains	Controller		
	Optimal with Complete Feedback	Extrapolated (step 0)	Final (step 10)
$K_{1,10}$	.685	.345	.005
$K_{1,12}$	.07228	.0723	.0720
$K_{1,13}$	.9918	.491	.491
$K_{1,14}$	- .6193	- .485	- .160
$K_{2,1}$	.008698	.00250	.00158
$K_{2,2}$	.004714	- .00850	- .00110
$K_{3,1}$	.001872	- .00073	.00013
$K_{3,2}$	.001119	.00072	.00633

Table 4. Performance of Incremental Gradient Algorithm[a]

λ	J (predicted gains)	J (corrected gains)
1.0	14377	14377
.9	15380	15186
.8	17375	16387
.7	18664	17479
.6	19394	18870
.5	25517	21308
.4	24179	23181
.3	26086	24832
.2	27202	25867
.1	28146	27194
0	30611	28015

[a]J for the free aircraft is 116, 193.

SECTION III
PARAMETER VARIATION CONSIDERATIONS

The goal of implicitly including parameter-variation constraints within quadratic optimization formulations motivated study in two directions. One phase of the study was aimed at discovering properties of the optimal performance surface over a segment of parameter space and utilizing such properties to find optimal insensitive controllers. The other phase of study was aimed at reducing sensitivity to parameter variations by introducing proper compensators into the controller. Promising results of both theoretical and practical value were obtained in the first phase. Although intuitively appealing, the problem formulation of the second phase appears to be of questionable practical value.

PROPERTIES OF THE OPTIMAL PERFORMANCE SURFACE

Consider the scalar equation $\dot{x} = fx + gu + \eta$ with performance functional $J = E \int_0^\infty (4x^2+u^2)dt$ where η is a unity white-noise input. The coefficients of the system, f and g, are constants with values in the rectangle R defined by $0 \leq f \leq 2$ and $1 \leq g \leq 5$. For a controller, $u = kx$, the performance functional is a function of the three parameters, k, f and g; i.e., $J = J(k, f, g)$. The optimal control for any (f, g) in R is $u = k^*(f, g)x$ where $k^*(f, g)$ minimizes $J(k, f, g)$. For this simple example, the solution for $k^*(f, g)$ is easily determined to be $k^*(f, g) = [-f- \sqrt{f^2 + 4g^2}]g^{-1}$. In this case $J(k^*(f, g), f, g)$ can be obtained explicitly as $-k^*(f, g)g^{-1}$. This surface over the rectangle R, as depicted in Figure 6, has a maximum over the corner (2, 1) and a minimum over the opposite corner (0, 5). Further calculations show that the surface $J(k^*(1, 3)f, g)$ corresponding to performance with the optimal controller for the midpoint of R highly accentuates the peak at (2, 1) while providing nearly optimal

performance for points close to the midpoint of R as shown in Figure 7. This controller appears to be sensitive to parameter variations. To alleviate this sensitivity, a controller could be chosen to minimize $E_{f,g}\{J(k, f, g)\}$ assuming a uniform probability distribution for the parameters f and g in the rectangle R. With this simple example, this can be accomplished yielding a value of $\bar{k} = -2.82$. Indeed the peak of $J(\bar{k}, f, g)$ which occurs at (2, 1) is lower, as shown in Figure 8. But the controller which minimizes the maximum of $J(k, f, g)$ over R is the optimal controller for the point (2, 1). Any other k will give a higher value of $J(k, 2, 1)$. Furthermore $J(k^*(2, 1)f, g)$ is less than or equal to $J(k^*(2, 1), 2, 1)$ as shown in Figure 9. This controller is the least sensitive to parameter variations defined by R.

The following two theorems indicate that the phenomenon noted in the above example is not mere happenstance. The first theorem is an existence theorem providing a sufficiency condition for optimal insensitive controllers locally, that is, for sufficiently small variations. This theorem also implies that such controllers are optimal controllers corresponding to boundary points of admissible parameter variation sets. The second theorem states a necessary condition for an optimal controller corresponding to a point in the boundary of the domain of admissible parameter variations to be an optimal insensitive controller. Here, the expression optimal insensitive controller refers to a controller which is optimal for some admissible value of the parameters which minimizes the maximum of the performance over the range of admissible parameter values.

- Theorem 1 — Consider the system $\dot{x} = F(p)x + G(p)u$, $x(0) = x_o$ and an associated performance functional $J(u, p) = \int_o^\infty (Hx+Du)^T Q(Hx+Du)dt$ where p is a vector of parameters. Let $J^*(p) = \min_u J(u, p)$. Suppose p_o is a point with the property that $\nabla_p J^*(p_o) \neq 0$. Then there exists an $\epsilon > 0$ such that the control $u^*(p_o)$ which minimizes $J(u, p_o)$ also minimizes the maximum of $J(u, p)$ with respect to p in an ϵ-ball with p_o on the shell (boundary) of the ball.

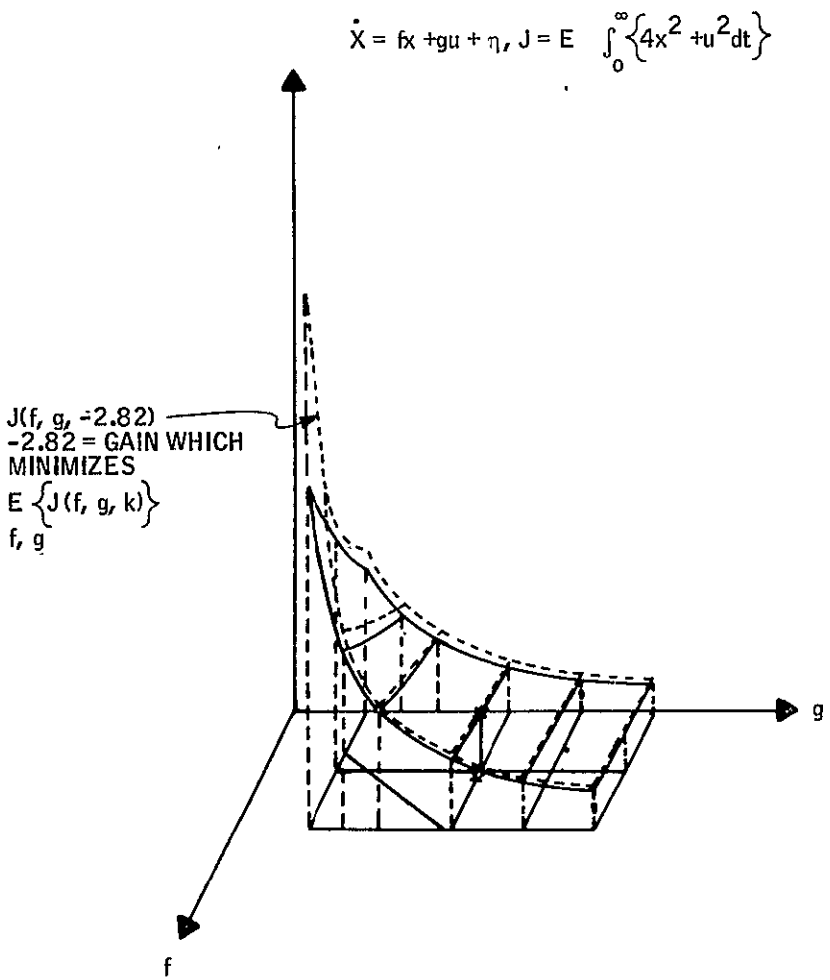


Figure 8. Performance Surface for Optimized Expected Parameters

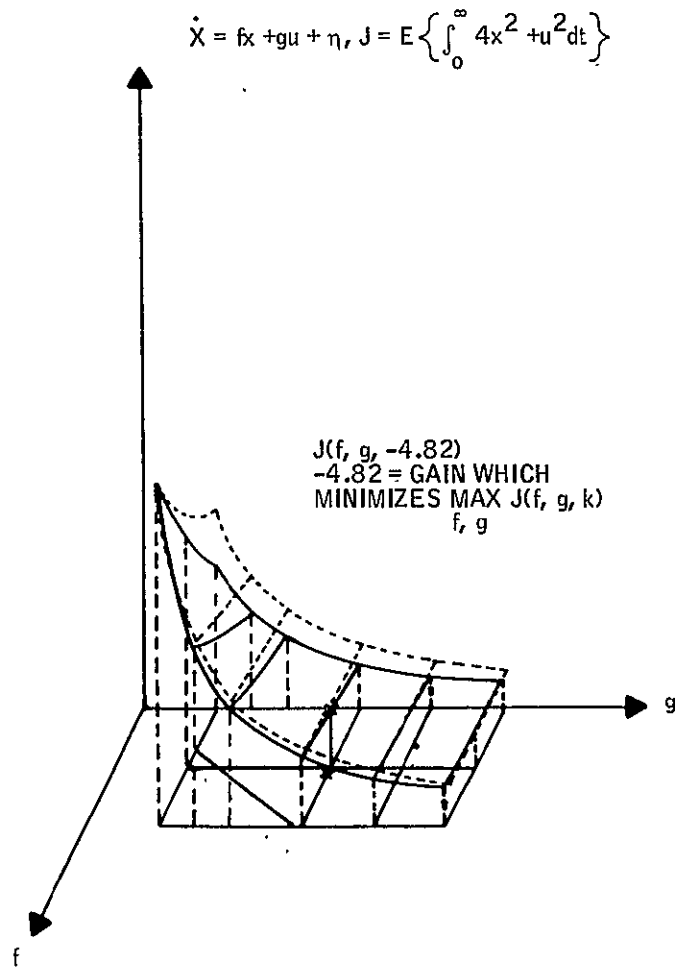


Figure 9. Performance Surface for Optimal Insensitive Controller

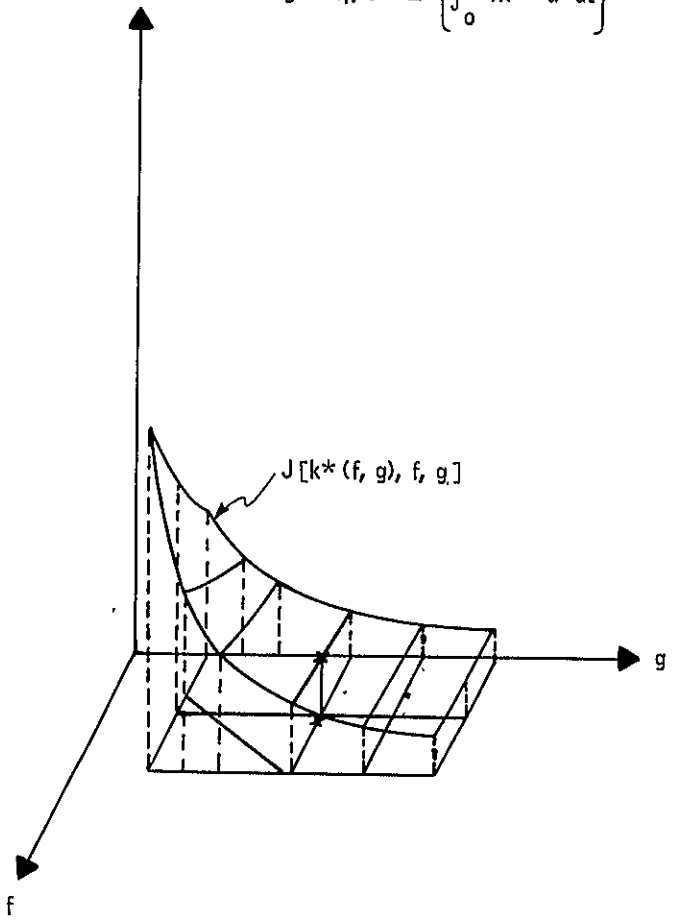


Figure 6. Optimal Performance Surface

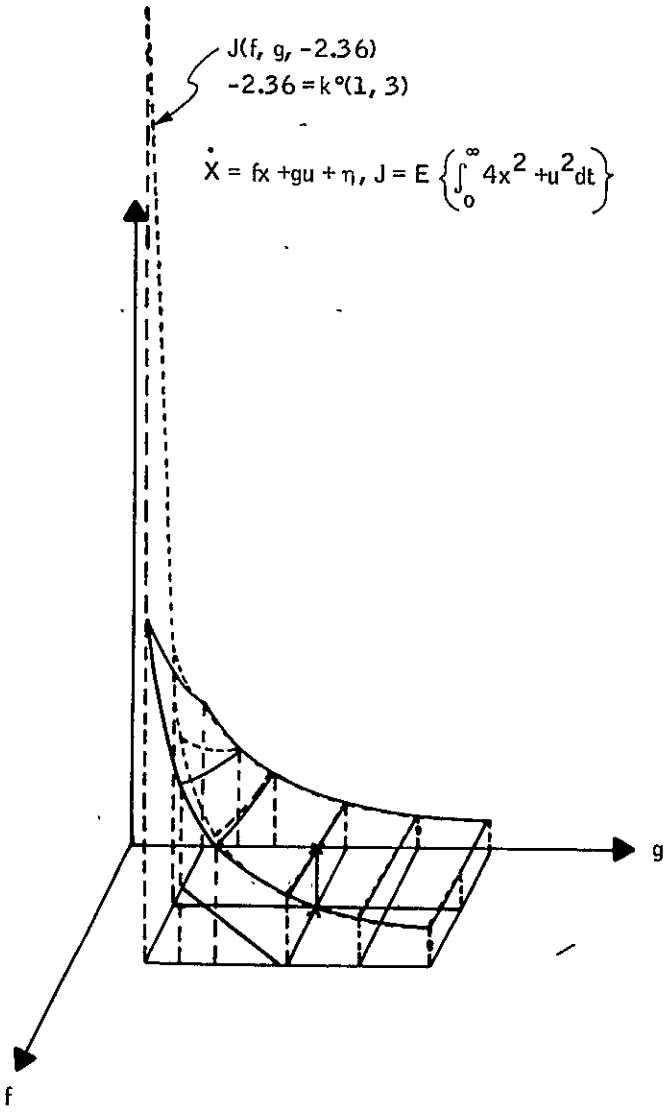


Figure 7. Nominal Performance Surface

Proof: For any $\epsilon > 0$ let $B_\epsilon(p_o)$ denote the ϵ-ball with center at $p_o - \epsilon \nabla_p J^*(p_o)$, i.e.,

$$B_\epsilon = \{p : p = p_o - \epsilon \nabla_p J^*(p_o) + \epsilon \eta, \quad |\eta| \leq |\nabla_p J^*(p_o)|\}.$$

Also define $M(u; p_o, \epsilon)$ to be $\max_{p \in B_\epsilon(p_o)} J(u, p)$. Then

$M(u; p_o, \epsilon) \geq J(u, p_o) \geq J^*(p_o)$. For $p \in B_\epsilon(p_o)$ we may express $J(u^*(p_o), p)$ as

$$J(u^*(p_o), p) = J(u^*(p_o), p_o) + \nabla_p J(u^*(p_o), p)\Big|_{p=p_o} \cdot (p-p_o) +$$

$$1/2(p-p_o)^T \left.\frac{\partial^2 J(u^*(p_o), p)}{\partial p^T}\right|_{p=p_o} (p-p_o) + o(\epsilon)^2$$

$$\equiv J^*(p_o) + g \cdot (p-p_o) + 1/2(p-p_o)^T H(p-p_o) + o(\epsilon^2), \; H > 0 \tag{32}$$

Note that $g \equiv \nabla_p J(u^*(p_o), p)\Big|_{p=p_o} = \nabla_p J^*(p)\Big|_{p=p_o}$

For $p \in B_\epsilon$ and ϵ sufficiently small the only possibilities for extreme points of $J(u^*(p_o), p)$ are

(1) approximately $p_o - H^{-1}g$ if this point lies within B_ϵ, or

(2) points on the shell of B_ϵ.

The point near $p_o - H^{-1}g$ is a minimizing point. Therefore maximizing points lie on the boundary of B_ϵ.

The problem of extremizing $J(u^*(p_o), p)$ subject to $p = p_o - \epsilon g + \epsilon\eta$ with $|\eta| = |g|$ may be treated with a Lagrange multiplier as minimizing

$$H = J_o + g \cdot (p-p_o) + 1/2(p-p_o)^T H(p-p_o) + o(\epsilon^2) + \lambda(\eta \cdot \eta - g \cdot g).$$

This yields $0 = \epsilon g + \lambda\eta - \epsilon^2 H(g-\eta) + 0(\epsilon^2)$ and $|\eta| = |g|$ as necessary conditions. For ϵ small these conditions imply that

$$\lambda = \pm\epsilon[1 + o(1)] \text{ and } \eta = \mp g[1 + o(1)].$$

The bottom signs yield the maximum and the top signs yield the minimum. The exact solution for the bottom signs is $\lambda = -\epsilon$, $\eta = g$ which describes the point p_o. Thus on B_ϵ, $J(u^*(p_o), p) \leq J(u^*(p_o), p_o) = J^*(p_o)$. Hence

$$M(u^*(p_o); p_o, \epsilon) = J^*(p_o) \leq M(u; p_o, \epsilon)$$

which was to be proved.

Theorem 2 — Consider $J(u^*(p_o), p) = J^*(p_o) + g \cdot (p-p_o) + o(|p-p_o|)$ with $g = \nabla_p J^*(p)|_{p=p_o}$. Let Ω denote a closed convex set with non-empty interior in the parameter space. Suppose p_o is a point in the boundary of Ω with the property that

$$J(u^*(p_o), p_o) = \max_{p\in\Omega} J(u^*(p_o), p).$$

Then g must be an external normal to Ω at p_o.

Proof: Assume that g is not an external normal to Ω at p_o, i.e., there exists a p_1 in Ω such that $(p_1-p_o) \cdot g = c_1 > 0$. Since Ω is convex $p(\lambda) = p_o + \lambda(p_1-p_o)$ lies in Ω for $0 \leq \lambda \leq 1$ and $(p(\lambda)-p_o) \cdot g = \lambda (p_1-p_o) \cdot g = \lambda c_1$. Thus

$$J(u^*(p_o), p(\lambda)) = J^*(p_o) + (p(\lambda)-p_o) \cdot g + o(|p(\lambda)-p_o|)$$

$$= J^*(p_o) + \lambda c_1 + o(\lambda) > J^*(p_o) \tag{33}$$

for λ sufficiently small. This contradicts the hypothesis that p_o has the property that $J(u^*(p_o), p_o) = \max_{p \in \Omega} J(u^*(p_o), p)$.

To utilize information about performance surfaces in determining insensitive controllers, approximations and computational techniques are required. For this purpose, consider the system

$$\dot{x} = F(p)x + G(p)u, \quad x(0) = x_o$$

and a performance index

$$J = \int_o^\infty (Hx+Du)^T Q(Hx+Du)dt$$

Suppose $J^*(p)$ denotes the optimal performance surface, $J(K^*(p), p)$, corresponding to $u = K^*(p)x$. For an arbitrary controller $u = Kx$, let $J(K, p)$ denote the corresponding cost surface. Define the "error" index as the difference

$$e(K, p) \triangleq J(K, p) - J^*(p)$$

This error function may be expressed as an implicit function of K and p as

$$e(K, p) = Tr\{AX_o\} - Tr\{PX_o\} \tag{34}$$

where $X_o = x_o x_o^T$; A satisfies

$$(F+GK)^T A + A(F+GH) + (H+DK)^T Q(H+DK) = 0$$

and P satisfies

$$F^T P + PF + H^T QH - PG(D^T QD)^{-1} G^T P - PG(D^T QD)^{-1} D^T QH$$
$$- H^T QD(D^T QD)^{-1} G^T P - H^T QD(D^T QD)^{-1} D^T QH = 0$$

A is an implicit function of K and p and P is an implicit function of only p. So J(K) and J*(p) can be approximated by truncated Taylor series expansions. It is assumed that the parameter variations of F and G are moderate, say 10 or 20 percent variation from the nominal value. Therefore, we might not need higher-order partials of A and P with respect to K and p. At most, it is assumed sufficient to expand J and J* in Taylor series up to the second-order partials of A and P. It might turn out for a practical system that the first-order expansion of J and J* would be sufficiently accurate. In any case the error function e(K, p) can be explicitly expressed in terms of K and p in a reasonably accurate form. Its derivation is given in Appendix D. A simple geometrical interpretation of the approximations of J and J* is shown in Figure 10, where p is a one-dimensional parameter, and K is fixed.

In Appendix D the following facts are established.

(1) $e(K^*(p), p) = 0$

This implies that the surfaces J*(p) and J(K*(p), p) touch at p.

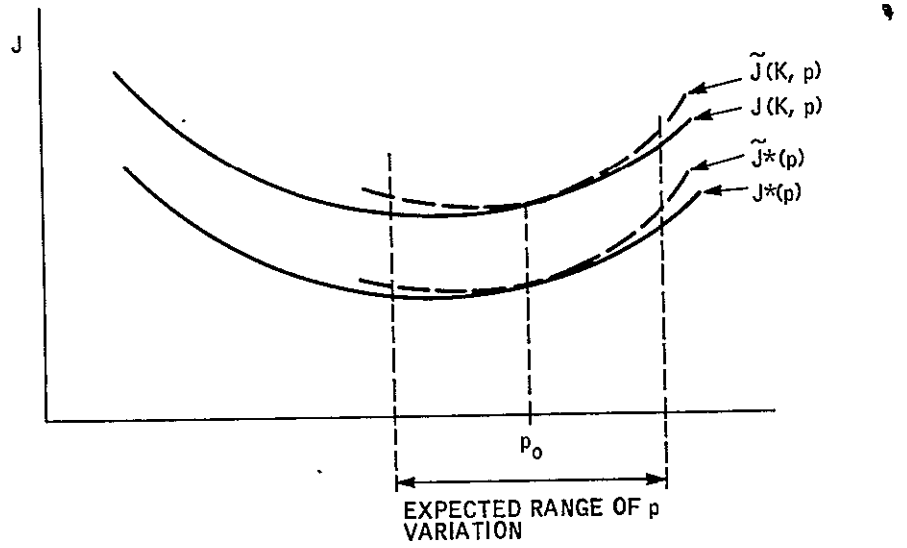


Figure 10. A Geometrical Interpretation of Taylor Series Approximations of J(K, p) and J*(p)

(2) $\nabla_p e(K^*(p), p) = 0$

This implies the surfaces are tangent at p.

(3) $\nabla_K e(K^*(p), p) = 0$

This is the statement that K*(p) satisfies the first-order condition of optimality.

(4) $\nabla_{KK} e(K^*(p), p) > 0$

This, together with (3) states that K*(p) is optimal.

These properties along with computational techniques described in Appendix D for obtaining truncated Taylor Series approximations make it reasonable to compute optimal insensitive controllers for realistic systems.

Results similar to those above were derived by Salmon (Ref. 20).

As a test vehicle for this approach, the C-5A model was modified to include only rigid-body and three flexure degrees of freedom and first-order actuator dynamics. Variations in the natural frequencies of the flexure modes were introduced as the parameter variations. Numerical results are presented in Appendix E.

INSENSITIVITY VIA COMPENSATORS

This phase of the study was aimed at relating the structure of the controller to the sensitivity of the controlled system. For a system represented by the familiar equations

$$\dot{x} = Fx + G_1 u + G_2 \eta$$

$$r = Hx + Du$$

and the associated performance functional

$$J = E\left\{\int r^T Q r \, dt\right\}$$

where the matrices F, G_1, G_2, H, D and Q may depend on a vector of parameters, p, it is known that the optimal controller is given by $u = Kx$ where K may depend on p. The equation $u = Kx$ has a simple geometric interpretation. The equation defines a subspace of the (x, u) space. With x and u scalars this interpretation is reduced to: $u = Kx$ defines a line in the (x, u) plane. Then optimality would be equivalent to motion in the (x, u) plane being constrained to the proper line, $u = Kx$, and (for the regulator problem) that undisturbed motion be directed toward the origin. Thus for a simple scalar system optimality might be depicted as in Figure 11 where the "optimal" lines for three values of p are sketched.

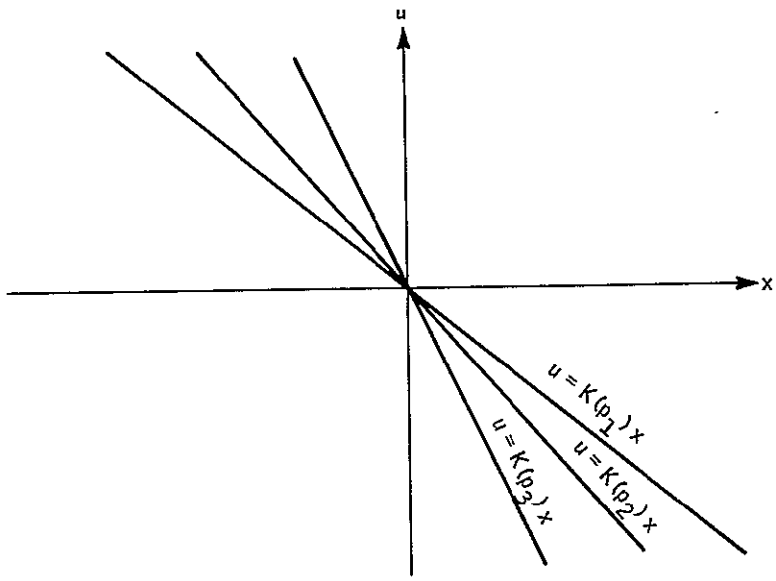


Figure 11. Optimal Motion in the (x, u) Plane

Compensation of the feedback may be represented in differential equation form by augmenting the equations with

$$u = Kx + K_o v$$

$$\dot{v} = w$$

where w is treated as the control input and the state vector is augmented to include v. For the augmented system, optimality may be approximated by asking that the motion of the augmented system by approximately in the "optimal" subspace. Then a performance criterion could take the following form: motion originating in the "optimal" subspace should remain in that subspace and be appropriately stable, motion originating outside the "optimal" subspace should approach the "optimal" subspace in an appropriate manner. In differential-equation terminology, this criterion can be expressed simply as requiring the "optimal" subspace to be a stable invariant manifold.

The single scalar system $\dot{x} = Fx + Gu$, $x(0)$ given, with performance index $J = \int_0^\infty (Qx^2+u^2)dt$ was analyzed to gain insight into the effectiveness of compensator structure in alleviating sensitivity to parameter variations. For numerical calculations it was assumed that $F = 1+2p$, $G = 3+4p$, $Q = 2.25$ and that $-0.6 \leq p \leq +0.6$.

The optimal control for any fixed value of p is $u = K^*(p)x$ with

$$K^* = -[F + \sqrt{F^2 + QG^2}]G^{-1}$$

The desired behavior of the controlled system was assumed to be that a closed-loop system principal axis match the line $u = K^*(p)x$ to a high degree. The Laplace transform of the system with $w = K_1x + K_2v$ is (without loss of generality K_o is set equal to 1):

$$\begin{aligned} (s-F)X(s) &= x(0) + G[KX(s)+V(s)] \\ (s-K_2)V(s) &= v(0) + K_1X(s) \\ U(s) &= KX(s) + V(s) \end{aligned} \tag{35}$$

From these equations

$$\frac{V(s)}{X(s)} = K + \frac{-x(0)K_1 + v(0)(s-F-GK)}{x(0)(s-K_2) + Gv(0)}$$

The requirement that $u(t) \equiv K^*x(t)$ if $u(0) = K^*x(0)$ yields the equation:

$$K_1 + (K_2-F-GK)(K^*-K) - G(K^*-K)^2 = 0$$

The left-hand side may be expanded as a Taylor Series in p about $p = 0$ yielding

$$K_1 + (K_2-1-3K)[K^*(0)-K] - 3[K^*(0)-K]^2$$

$$+ p\{-(2+4K)[K^*(0)-K] - 4[K^*(0)-K]^2 + (K_2-1-3K-6[K^*(0)-K])[K^*(0)]'\}$$

$$+1/2p^2\{(K_2-1-3K-6[K^*(0)-K])[K^*(0)]'' - 6([K^*(0)]')^2$$

$$-(2+4K+8[K^*(0)-K])[K^*(0)]'\} + \text{higher order terms} \qquad (36)$$

The control parameters K, K_1 and K_2 can be chosen so that the constant term and the coefficients of p and p^2 are zero to yield desired matching of the principal axis to $u = K^*(p)x$ for p near zero. Stability requirements of the resulting closed-loop system are not guaranteed. This may impose further constraints on the control parameters and, in so doing, reduce the degree of approximation.

For the numerical example it was found that matching to third-order terms in p could be achieved with K = -2.08, K_1 = -1.03 and K_2 = 0.29. The corresponding closed-loop poles for p = 0 are -4.61 and -0.34. The pole at -4.61 is the desired value for the optimally controlled scalar system with p = 0. Thus the compensator pole now is the dominant pole, and the corresponding principal coordinate displays the dominant dynamics. This is undesirable. The principal axis which approximates $u = K^*(p)x$ is an unstable axis. Motion originating near this axis does not remain close to this axis. Figure 12 is a sketch showing this kind of behavior.

One other case was considered in which K was set equal to zero. The remaining control parameters K_1 and K_2 were chosen to eliminate the constant and first-order terms in p. The value of the gains were K_1 = -8.15 and K_2 = -48.2. The corresponding closed loop poles for p = 0 were -4.61 and -42.7. In this case the axis $u = K^*x$ was stable and the resulting controller exhibited desired properties. Characteristics of this controller are given in Table 5. The final column of this table characterizes the

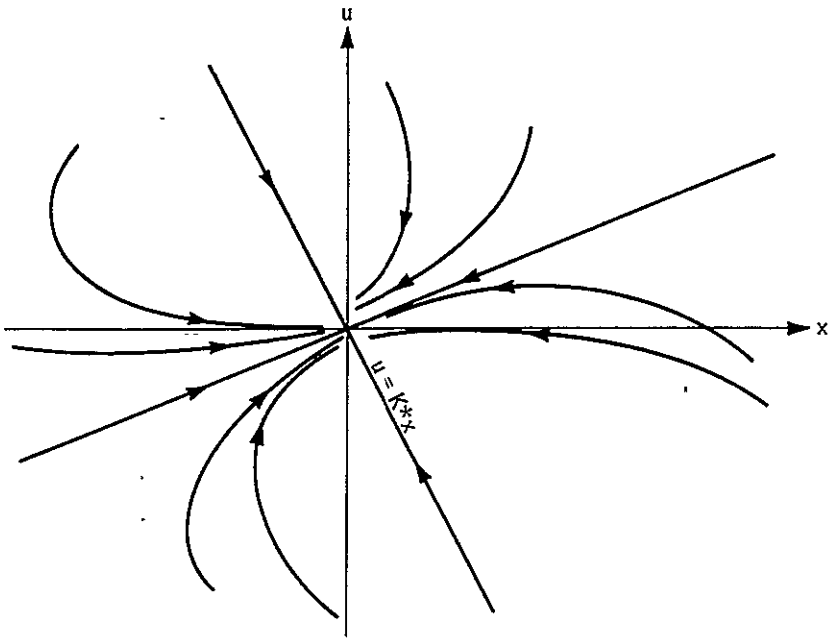


Figure 12. Phase-Plane Portrait Indicating Unstable Subspace u = K*x

behavior of the optimal controller based on p = 0 for comparison. Columns 2 and 3 indicate good correlation between the dominant principal axis and the line u = K*(p)x. Columns 4, 5 and 8 show eigenvalues with good matching between columns 4 and 5. On this basis the compensated system appears to be less sensitive than an uncompensated system. However, the improvement may not be worth the added complexity.

Thus, although this method of alleviating sensitivity has some appeal, its utility would have to be determined by application to a more realistic and more complicated example. Rather than pursue this, it was considered more advantageous to study the performance surface for a realistic example.

Table 5. Optimal Control and Principal Coordinate Dependence on p

1	2	3	4	5	6	7	8
p	K*	q_1†	F + GK*	μ	$-q_2$†	λ	F + GK*(0)
+.6	-1.96	-2.08	-8.39	-9.03	- 7.3	-37.0	-7.90
+.4	-1.94	-2.00	-7.13	-7.38	- 8.9	-39.1	-6.80
+.2	-1.91	-1.93	-5.87	-5.92	-11.1	-41.0	-5.71
+.1	-1.89	-1.90	-5.23	-5.25	-12.7	-41.9	-5.16
0	-1.87	-1.87	-4.61	-4.61	-14.6	-42.7	-4.61
-.1	-1.84	-1.84	-3.98	-4.00	-17.0	-43.5	-4.06
-.2	-1.80	-1.82	-3.36	-3.40	-20.4	-44.3	-3.51
-.4	-1.65	-1.78	-2.11	-2.29	-32.8	-45.8	-2.42
-.6	-1.20	-1.74	- .92	-1.24	-78.1	-47.2	-1.32

†Principal coordinates are $z_1 = x_2 + q_1 x_1$ and $z_2 = x_2 + q_2 x_1$

SECTION IV
SENSOR CHOICE

An important problem in the design of practical controllers for large flexible vehicles is the choice of types and locations of sensors to generate the feedback signals. The problem arises because it is economically prohibitive in most applications to sense all system states, particularly the rates and displacements of numerous flexure modes. Moreover, experience on previous flexure control designs has shown that adequate performance can be obtained with significantly fewer sensors than state variables, suggesting that economics vs. performance comparisons favor smaller sensor complements.

In principle, the sensor-choice problem is solved by selecting a complement of instruments which exhibits the most desirable cost/performance tradeoff, assuming, of course, that the instruments are utilized in an optimum fashion, i.e., by locating them optimally along the vehicle and by using optimally the information thus provided. It is in these areas that present methodology fails. We have too little computing power, or too cumbersome computing methods, to determine optimal locations of sensors and to utilize their information optimally with a controller subject to various simplicity constraints. Cost/performance comparison for several competing sensor complements are thus difficult to come by.

The research reported in this section is intended to improve this situation. Its general objectives are twofold: (1) to increase understanding of the relationships between sensor complements and the performance capabilities they offer and (2) to improve computational methods for locating and utilizing the instruments.

MATHEMATICS OF THE SENSOR-CHOICE PROBLEM

As usual, we are given a mathematical model of the following form:

- System:

$$\dot{x} = F(t)x + G_1(t)u + G_2(t)\eta \tag{37}$$

- Responses:

$$r = H(t)x + D(t)u \tag{38}$$

- Performance Measure:

$$J = E\left\{ \int_0^T r^T(t)\, Qr(t)\, dt \right\} \tag{39}$$

where all symbols correspond to previous definitions. Then we are given a collection of measurement instruments (rate and acceleration sensors, stress sensors, air data sensors, etc.) from which we may select any number and combination to mount anywhere on the airframe. The outputs of these instruments are to be used for control. That is, let Ω denote a particular subset of instruments, and let vector y denote the locations of each instrument (i.e., y_i is the location of the i^{th} sensor). Each location can be represented by a scalar number if fuselage and wing positions are laid end to end. Then the instrument outputs, at least for linearized models, will be

$$z = M(t, y, \Omega)x \tag{40}$$

and the controller must be of the form

$$u(t) = L(t, z) \tag{41}$$

In these latter equations, $M(t, y, \Omega)$ denotes a matrix of dimension (mxn), where m is the number of sensors in the set Ω, and the function $L(t, z)$ is a linear control law which is usually constrained to exhibit certain attributes of simplicity. For instance, the nondynamic form, $L(t, z) = K(t)z$, or a dynamic form with low-order compensation are desirable.

With measurements (40) and controller (41) substituted into equation (37), the performance measure (39) becomes a function of Ω, y and L, i.e.,

$$J = J(\Omega, y, L) \tag{42}$$

This expression shows explicitly why cost/performance comparisons of sensor complements are difficult to get. To evaluate the performance capability, $J^*(\Omega)$, of the set of instruments Ω (whose cost at a given level of reliability is presumably known), we must specify both the sensor locations and the control law which is best for Ω, i.e.,

$$\begin{aligned} J^*(\Omega) &= \min_{y} \min_{L} J(\Omega, y, L) \\ &= J(\Omega, y^*(\Omega), L^*(.,.,\Omega)) \end{aligned} \tag{43}$$

With current capabilities, the simultaneous optimizations of y and L, with L subject to simplicity constraints, are expensive indeed. Just to get a feel for the magnitude of the problem, suppose we treat a single fixed-time point of the general nonstationary situation. Suppose further that a nondynamic controller of the form $L(z) = Kz$ is sought. Then we could use one of the parametric methods in Section II to compute optimal controllers, L, as functions of y and Ω and we could imbed these computations inside an iterative Newton-Raphson or gradient loop to optimize y. Diagrammatically, the procedure would like like Figure 13.

Depending upon the order and complexity of the problem, the inner loop of this algorithm [block (1)] consumes anywhere from one-half to three hours of computing, say on an SDS 9300 machine.† The outer loop updating step

†The one-half-hour figure was obtained on a 20th-order single flight condition optimization for the F4 aircraft (Ref. 6), while the three-hour figure was obtained for the 23rd-order C-5A example discussed in Section II.

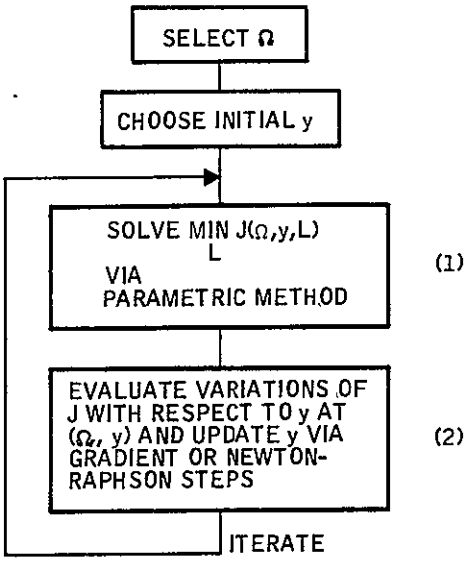


Figure 13. Basic Sensor-Choice Algorithm

[block (2)] can be performed either by repeated executions of block (1) at perturbed values of y or by the solution of coupled state covariance and costate equations which result from analytic evaluations of the first and second variations of J. For first partials alone, the latter equations comprise a system of n(n-1) linear equations. The computations in both blocks are thus extremely time consuming — and, of course, they must be repeated for as many outer-loop iterations as are required for convergence.

While it is thus apparent that the algorithm of Figure 13 is ill-suited to the general sensor-choice problem, it should be noted that it has been seriously proposed for more specialized versions of the problem (Ref. 11) and has been successfully run on a specialized version of the "dual" problem — force producer choice (Ref. 5). In each case, the specializations were such as to greatly reduce the computational loads of blocks (1) and (2). Namely, it was assumed that controller L, is subject to no practicality constraints. This leads in block (1) to the controller $L^*(\Omega, y)$ which consists of a cascaded Kalman filter and Kalman controller (a system with n^{th} order compensation).

Both the filter and controller are comparatively easy to compute. Moreover, the variations in block (2) also turn out to be simpler. This is because the Kalman controller is independent of y in the case of sensor choice, and the Kalman filter is independent of y in the case of force producer choice. The previously coupled covariance and costate equations uncouple and leave comparatively simple equations for the variations of J.

Since we are concerned here specifically with constrained controllers, the simplifications offered by these special cases are inapplicable and the algorithm of Figure 13 offers little hope toward solving the sensor-choice problem efficiently. The research reported here has therefore been directed along different lines. Instead of relying on the quadratic cost, $J^*(\Omega)$, as a performance measure of a sensor complement, the research was aimed toward development of alternate quality measures with two key properties:

(1) They should be easy to evaluate, for use in computational algorithms such as Figure 13.

(2) They should provide meaningful indications of performance capabilities offered by a set of sensors. Ideally, they should exhibit strong correlation with the cost $J^*(\Omega)$.

This line of attack follows the approach taken in Reference 3, where two alternate quality measures were proposed, though neither proved wholly successful in satisfying property (2).

Like Reference 3, the present research has no final wholly successful answers to report. Two promising quality measures were developed, both based on the pole-placement capabilities of the sensor complement. As such, they strictly apply only to stationary problems, but can be used on nonstationary ones via frozen-point procedures. Most of the development work has been concerned with establishing theoretical equations and verifying the computational characteristics required by property (1).

Results of these efforts are described below. They are general in nature and find application outside of the sensor choice problem in areas such as multiloop controller design, initiation of computational algorithms, and sensitivity analysis. As far as property (2) is concerned, several basic performance capabilities (e.g., stability, modal characteristics) follow directly from the measures. Correlations with $J^*(\Omega)$, however, and the practical utility of the measures remain to be established on realistic design problems.

POLE-PLACEMENT QUALITY MEASURES

The basic notion pursued here is to examine the freedom offered by a sensor complement in locating closed-loop eigenvalues as a possible source of quality measures. This is motivated by three considerations. First, classical experience with root loci and frequency domain design techniques provides tested insightful relationships between the performance capabilities of a controlled system and the closed-loop pole arrangements permitted by sensors. Such notions as stability, frequencies of oscillation, damping of individual modes of response and dominance are all apparent from the pole constellation. Second, there is a fundamental connection between pole placement and the concept of controllability. With full state feedback, freedom to assign all n system poles arbitrarily has been shown mathematically equivalent to the condition of complete controllability (Ref. 12, 13). Hence it appears fruitful to investigate any restrictions of this pole assignability property which are imposed by partial state feedback. Third, in certain special control problems, namely single input, the state dependent terms of the quadratic cost (39) can be expressed uniquely in terms of closed loop poles, with no dependence on gains or measurements used to bring them about. Though this property fails for multi-input systems and ignores control-dependent terms, it provides a potential link between quality measures based on pole placement and the measure $J^*(\Omega)$.

As shown in References 14 and 15, the primary effect of limited state measurement is to reduce the total number of poles which can be assigned arbitrarily – from the maximum of n poles with full state measurement to a maximum number, p_{max}, which will be defined later. The effect does not confine each pole to some subregion of the complex plane, as one might expect from classical root locus plots. Rather, p_{max} poles may be placed anywhere in the plane (subject to certain nonsingularity constraints), with the remaining $(n-p_{max})$ poles implicitly determined. These observations hold for direct feedback of the measurement with a nondynamic controller $L = Kz$. If compensation is allowed, the number of arbitrary poles can be increased by an amount also defined later.

These properties suggest the following quality measures for sensor complements:

- The number of poles, p_{max}, which can be placed arbitrarily
- Some measure of deviation (say quadratic) of the unassignable poles from specified desirable positions, given that the assignable poles are placed in desired spots

While these are only verbal definitions of what must eventually be analytic measures, their potential is apparent. The first serves as a gross differentiator between competing sensor complements. If performance requires that p poles be positioned accurately, then all combinations of instruments which fail to have this capability can be eliminated immediately. Computations should be simple, since the measure depends only weakly on sensor locations, y. The second measure provides finer differentiation between the remaining complements. It evaluates the amount of pole placement deterioration in noncritical modes of response, given that the critical modes are adequately controlled. This measure should depend strongly on y and could thus be used to locate the sensors by means of an algorithm such as Figure 13. For

both measures, the pole constellations generated by quadratic-optimal unconstrained controllers could be used to define critical and noncritical modes of response and desirable pole locations.

All this is conditioned, of course, on computational feasibility — whether it is possible to determine the maximum number of arbitrary poles and to design the controller which achieves the desired critical closed-loop locations with computational efforts significantly below the $J^*(\Omega)$ requirements. Pole-placement equations and algorithms in the literature suggest an affirmative answer (Ref. 14, 15, 16, 17). However, no published procedures were found which completely solve the specific problems posed here. For a non-dynamic multi-input multi-output controller, for example, available results establish that $p_{max} = m$, where m is the rank of the measurement matrix, $M(\Omega, y)$. This number is based on analytical methods which reduce the system to single-input form before assessing pole assignability properties. In this study it has been shown that significantly more poles can actually be placed when the additional degrees of freedom offered by multiple inputs are utilized. Similarly, no figure seems available for the number of arbitrary poles added by a compensator of specified order, and in particular, no procedure was found to specify the minimum compensator order required to achieve arbitrary placement of all system poles and to compute the parameters of this compensator. These questions were answered in the course of this research. The answers take the form of a general set of pole-placement equations and a computational algorithm to solve them. Both are discussed in the remainder of this section. The algorithm provides a computationally reasonable way to evaluate the quality measures proposed above. This is demonstrated in Appendix G with a few trial computations for F-4 and C-5A flight control design problems.

GENERAL POLE-PLACEMENT EQUATIONS

The poles of a closed-loop stationary system (37) with nondynamic controller $L = Kz$ are roots of the following characteristic polynomial:

$$P(s) = D(s) + \sum_{k=1}^{\min(r,m)} \left\{ \sum_{j_1 \cdots j_k} \cdots \sum_{\ell_1 \cdots \ell_k} N_{\ell_1 \cdots \ell_k}^{j_1 \cdots j_k}(s) \left[\sum_{p_\ell} (\pm) K_{j_1 \ell_1} K_{j_2 \ell_2} \cdots K_{j_k \ell_k} \right] \right\} \tag{44}$$

Here $P(s)$ is the closed-loop polynomial, $D(s)$ is the open loop polynomial of (37) with $u=0$, and the expressions $N_{\ell_1 \cdots \ell_k}^{j_1 \cdots j_k}(s)$ are various polynominals associated with the feedback loops. The summations indexed by $j_1, j_2, \cdots j_k$ and $\ell_1, \ell_2, \ldots, \ell_k$ are carried out over all naturally ordered groups of k out of r controls and k out of m measurements, respectively, and the summation indexed by p_ℓ is carried out over all permutations of the sequence $\ell_1, \ell_2, \ldots, \ell_k$, with algebraic sign taken positive for even permutations and negative for odd.

Equation (44) is a polynomial in the complex variable s and in the mr gain variables K, with order n in s and min(m, r) in K. Its detailed derivation is left to Appendix F. Here we will present only interpretations of terms and some procedures for computing them. We will then use shorthand versions of the equation to derive necessary and sufficient conditions for placement of p poles and to develop explicit expressions for p_{max}. These will be shown to apply to dynamic as well as nondynamic controllers; they will be illustrated with a small example pole-placement problem; and finally, they will be used to devise a computerized pole-placement algorithm.

Interpretations and Definitions

The terms, $N^{j_1 \cdots j_k}_{\ell_1 \cdots \ell_k}(s)$ may be treated as generalized numerator polynominals of the stationary multi-input system (37) with multiple stationary outputs (40). For example, the collection of first-order terms (k=1) are the familiar numerators of the transfer function matrix

$$H(s) = -M(\Omega, y)\,(sI-F)^{-1}G_1 \qquad (45)$$

Each term can therefore be computed by replacing certain rows of the matrix (sI-F) with rows of the measurement matrix $M(\Omega, y)$ and evaluating the resulting determinant. For the term N^{j}_{ℓ}, this means replacement of the $i(j)^{th}$ row of (sI-F), call it $(sI-F)^{(i(j))}$, by $-(G_1)_{i(j)j}$ times the ℓ^{th} row, $M^{(\ell)}$, of matrix M. This procedure assumes that each column of matrix G_1 has a single nonzero entry in positions i(j), j = 1, 2, . . . , r; which will always be the case for systems with actuator dynamics and represents no loss of generality even for other cases.

The collection of second-order terms (k=2) are so-called "coupling numerators" of the system (Ref. 18) which are present whenever two feedback paths exist simultaneously. Each of these terms is computed by replacing two rows of (sI-F) with rows of M and computing determinants. For $N^{j_1 j_2}_{\ell_1 \ell_2}$, the replacement schedule would be $(sI-F)^{(i(j_1))}$ replaced by $-(G_1)_{i(j_1)j_1}M^{(\ell_1)}$ and $(sI-F)^{(i(j_2))}$ replaced by $-(G_1)_{i(j_2)j_2}M^{(\ell_2)}$.

Analogous replacement schedules apply for higher-order numerator terms, with k row replacements for each k^{th}-order term. No simple interpretation of these terms, however, seems to be possible except to say that they arise whenever k feedback paths exist simultaneously.

Conditions for Arbitrary Pole Placement

For ease of manipulation, it is convenient to express Equation (44) in the following shorthand form:

$$P(s) = D(s) + \sum_i N_i(s)\, \phi_i(K) \tag{46}$$

where the various generalized numerators are designated simply by $N_i(s)$ and the gain functions which multiply them are denoted by $\phi_i(K)$. It is now our objective to determine the maximum number of roots of the polynominal P(s) which can be assigned arbitrarily by proper choice of K and to determine the fate of the remaining roots. To do this, assume that the number of arbitrary poles is p, whose maximum is as yet unknown. Assign these poles to be roots of a specified polynomial $\lambda(s)$, and let the remaining poles be roots of the polynomial $\delta(s)$. Then (46) becomes

$$P(s) = \lambda(s)\, \delta(s) = D(s) + \sum_i N_i(s)\, \phi_i(K) \tag{47}$$

and in coefficient-vector-form† we get

$$\Lambda\delta + \bar{\lambda} = D + \sum_i N_i\, \phi_i(K)$$

$$0 = D - \bar{\lambda} - \Lambda\delta + \sum_i N_i\, \phi_i(K) \tag{48}$$

where Λ is a matrix and $\bar{\lambda}$ a vector defined as follows:

†In coefficient-vector-form, a polynomial $P(s) = p_1 + p_2 s + p_3 s^2 + \cdots p_n s^{n-1} + s^n$ is represented by the n-vector $P = (p_1\ p_2 \cdots p_n)^T$.

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ \lambda_2 & \lambda_1 & & 0 \\ \lambda_3 & \lambda_2 & & 0 \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \lambda_n & \lambda_{n-1} & & \lambda_1 \\ 1 & \lambda_n & & \lambda_2 \\ 0 & 1 & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \\ \cdot & \cdot & & \\ 0 & 0 & & \lambda_n \\ 0 & 0 & \cdots & 1 \end{bmatrix}, \qquad \bar{\lambda} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ \lambda_1 \\ \lambda_2 \\ \cdot \\ \cdot \\ \cdot \\ \\ \lambda_{n-1} \\ \lambda_n \end{bmatrix} \tag{49}$$

Equations (48) represent a set of n nonlinear equations of the form $f(\bar{\lambda}, \delta, K) = 0$, which define K and δ as functions of the specified polynomial λ, i.e., $K = K(\lambda)$ and $\delta = \delta(\lambda)$. According to the implicit function theorem (Ref. 8), these functions exist in a neighborhood of a point $(\lambda_o, \delta_o, K_o)$ if, and only if, the rank of the "Jacobian matrix"

$$\psi(\lambda_o, \delta_o, K_o) = \left[\begin{array}{c|c} \dfrac{\partial f}{\partial \delta^T} & \dfrac{\partial f}{\partial K^T} \end{array}\right]_{\lambda_o \delta_o K_o} \tag{50}\dagger$$

$$= \left[\begin{array}{c|c} -\Lambda_o & \sum_i N_i \left.\dfrac{\partial \phi_i}{\partial K^T}\right|_{K_o} \end{array}\right]$$

†This equation makes sense in vector-matrix notation only if K is written out as a vector. This is assumed.

equals n, and they are unique if in addition the matrix is square. Existence of solutions $K(\lambda)$, $\delta(\lambda)$ are, of course, the very conditions which must be proved to establish arbitrary placement of the p poles, $\lambda(s)$. We have thus arrived at the following basic result:

- Theorem 1 – Locally arbitrary placement of p poles is possible in a neighborhood of $(\lambda_o, \delta_o, K_o)$ if, and only if, rank $[\psi(\lambda_o, \delta_o, K_o)] = n$

This theorem can be used to determine p_{max} and also to examine pole placement conditions in a special neighborhood of interest $(\lambda_o, \delta_o, 0)$. To determine p_{max}, note that the rank of ψ is bounded by the inequalities

$$\text{rank}\left[\sum_i N_i \frac{\partial \phi_i}{\partial K^T}\right] \le \text{rank}\,[\psi] \le \text{rank}\left[\sum_i N_i \frac{\partial \phi_i}{\partial K^T}\right] + \text{rank}\,[\Delta] \qquad (51)$$

with the right-hand equality realized when all columns of Δ are independent of columns of $\sum_i N_i \frac{\partial \phi_i}{\partial K^T}$. Since rank $[\psi]$ must be n, this condition gives

$$n = \text{rank}\left[\sum_i N_i \frac{\partial \phi_i}{\partial K^T}\right] + (n - p_{max})$$

$$p_{max}\Big|_{K_o} = \text{rank}\left[\sum_i N_i \frac{\partial \phi_i}{\partial K^T}\Big|_{K_o}\right] \qquad (52)$$

Now consider the special neighborhood $(\lambda_o, \delta_o, 0)$, i.e., pole placement in the vicinity of the open-loop system. For this neighborhood, all partials $\frac{\partial}{\partial K^T}\phi_i(K)$ vanish save those for the first-order terms N_ℓ^j. This gives

$$p_{max}\Big|_{K_o = 0} = \text{rank}\left\{N_1^1\, N_2^1 \ldots N_m^1 N_1^2 \ldots N_m^r\right\} \tag{53}$$

or verbally, the maximum number of arbitrary poles in the vicinity of the open-loop system is equal to the rank of the matrix of coefficient vectors formed from the system's numerators. If we assume that the system is completely controllable from at least one of its inputs, say u_v, and that the matrix $M(\Omega, y)$ has rank m, then the following bounds are readily established for p_{max}:

$$m \leq p_{max}\Big|_{K_o = 0} \leq \min\,(n,\ mr) \tag{54}$$

Application to Systems with Compensators

These results apply verbatim to systems which include dynamic compensators. For example, suppose that q^{th} order dynamic compensation is permitted in the controller L(z). We then append q integrators to the original dynamic system, i.e.

$$\left.\begin{aligned} \dot{x} &= Fx + G_1 u + G_2 \eta \\ \dot{x}_{n+1} &= x_{n+2} + u_{r+1} \\ \dot{x}_{n+2} &= x_{n+3} + u_{r+2} \\ &\vdots \\ \dot{x}_{n+q} &= u_{r+q} \end{aligned}\right\} \tag{37a}$$

$$z = \begin{bmatrix} M(\Omega, y)\, x \\ x_{n+1} \\ x_{n+2} \\ \vdots \\ x_{n+q} \end{bmatrix} \tag{40a}$$

$$u = \begin{bmatrix} K^{11} & K^{12} \\ K^{21} & K^{22} \end{bmatrix} z \tag{41a}$$

Each integrator adds one input and one output to the original collection of inputs and outputs and thus adds additional $N_i(s)\ \phi_i(K)$ terms to equation (46). The new value p_{max} is again given by equation (52) and it follows from this equation that

$$p_{max}\Big|_{\substack{\text{No} \\ \text{compensation}}} \leq p_{max}\Big|_{\substack{q^{th}\text{ order} \\ \text{compensation}}} \tag{55}$$

This inequality assumes that the right-hand side is evaluated at

$$K_o = \begin{bmatrix} K_o^{11} & K_o^{12} \\ K_o^{21} & K_o^{22} \end{bmatrix}$$

and the left-hand side at $K_o = K_o^{11}$. If in particular we let $K_o^{11} = 0$ and choose K_o^{12} such the $(m+1)^{th}$ measurement is connected to the control u_v (for which the original system is completely controllable), then an analogy to (54) yields

$$m+q \leq p_{max}\Big|_{\substack{q^{th}\text{ order} \\ \text{compensation} \\ K_o = \begin{bmatrix} 0 & K^{12} \\ 0 & 0 \end{bmatrix}}} \leq \min\left[(m+q)(r+q),\ n+q\right] \tag{56}$$

Inequalities (54) and (55) hold in general for different orders of compensation. That is, p_{max} for q_1^{th}-order compensation equals or exceeds p_{max} for q_2^{th}-order compensation as long as $q_2 \leq q_1$. This fact can be used to determine the minimum compensator order required to place any prespecified number of poles, p, and in particular, the order required to place all system poles (including the compensator). We simply start with q=1, check the rank condition (52), then go to q = 2, then q = 3, etc., until the rank equals n+q.

The procedure will generate not only the compensator order but also a compensator design, via the function $K(\lambda)$ and $\delta(\lambda)$ implied by equation (48).

All this is best illustrated with an example:

Consider the following system

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -d_1 & -d_2 & -d_3 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u \qquad \text{note: } \{i(1), \ldots, i(r)\} = \{i(1)\} = \{3\}$$

$$z = \begin{bmatrix} 1 & 0 & 0 \\ & & \\ 0 & 1 & 0 \end{bmatrix} x$$

Suppose we want to assess pole placement capability without compensation and to design a minimum order compensator (if needed) for arbitrary placement of all system poles.

Without compensation, we can use equation (44) directly to get the following first-order polynomial (in K) for the characteristic equation

$$(s^3 + p_3 s^2 + p_2 s + p_1) =$$

$$(s^3 + d_3 s^2 + d_2 s + d_1) + K_{11} \det \begin{bmatrix} s & -1 & 0 \\ 0 & s & -1 \\ -1 & 0 & 0 \end{bmatrix} + K_{12} \det \begin{bmatrix} s & -1 & 0 \\ 0 & s & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

or, in coefficient-vector-form

$$\begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} + K_{11} \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix} + K_{12} \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}$$

Applying (52) we find that

$$p_{max} = \text{rank} \begin{bmatrix} -1 & 0 \\ 0 & -1 \\ 0 & 0 \end{bmatrix} = 2$$

which means that only two poles can be assigned arbitrarily. The third remains beyond our control. Equations for all three can be obtained from equation (48) and the implicit function theorem, i.e.

specified roots: $s^2 + \lambda_2 s + \lambda_1$

unspecified root: $s + \delta_1$

$$\begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ 1 \end{bmatrix} \delta_1 + \begin{bmatrix} 0 \\ \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} + K_{11} \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix} + K_{12} \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} \delta_1 \\ K_{11} \\ K_{12} \end{bmatrix} = - \begin{bmatrix} -\lambda_1 & -1 & 0 \\ -\lambda_2 & 0 & -1 \\ -1 & 0 & 0 \end{bmatrix}^{-1} \begin{bmatrix} d_1 \\ d_2 - \lambda_1 \\ d_3 - \lambda_2 \end{bmatrix}$$

Now let's explore arbitrary placement of all system poles via compensation. We begin by appending a single integrator, working up to two, three, or more only if required. The augmented system is

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -d_1 & -d_2 & -d_3 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} u$$

$$z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x$$

with $\{i(1), \ldots, i(r)\} = \{i(1), i(2)\} = \{3, 4\}$

Equation (44) now becomes a second-order polynomial in K with first-order terms for multiplier functions $\phi_i(K) = K_{11}, K_{12}, K_{13}, K_{21}, K_{22}, K_{23}$ and second-order terms for $(K_{11}K_{22} - K_{12}K_{21})$, $(K_{11}K_{23} - K_{13}K_{21})$, and $(K_{12}K_{23} - K_{13}K_{22})$.

The result

$$\begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{bmatrix} = \begin{bmatrix} 0 \\ d_1 \\ d_2 \\ d_3 \end{bmatrix} + K_{11} \begin{bmatrix} 0 \\ -1 \\ 0 \\ 0 \end{bmatrix} + K_{12} \begin{bmatrix} 0 \\ 0 \\ -1 \\ 0 \end{bmatrix} + K_{23} \begin{bmatrix} -d_1 \\ -d_2 \\ -d_3 \\ -1 \end{bmatrix} +$$

$$+ (K_{11}K_{23} - K_{13}K_{21}) \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} + (K_{12}K_{23} - K_{13}K_{22}) \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

with all other terms equal to zero. Note, in particular, that all first order coupling terms between system and compensator vanish, i.e., terms for K_{13}, K_{21}, K_{22}. Again applying (52) gives

$$p_{max} = \text{rank} \begin{bmatrix} K_{23} & 0 & -K_{21} & -K_{13} & 0 & -d_1+K_{11} \\ -1 & K_{23} & -K_{22} & 0 & -K_{13} & -d_2+K_{12} \\ 0 & -1 & 0 & 0 & 0 & -d_3 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}$$

whenever $K_{13} \neq 0$ (this can be shown by computing a determinant for columns 2, 4, 5 and 6). Thus, arbitrary placement of all poles is possible with a first-order compensator in all neighborhoods except those about K_o such that $K_{13} = 0$. This merely says that the compensator will do no good unless its output is fed to the input of the original system. [This corresponds to the choice of K^{12} made to get equation (56).]

The compensator design and direct controller gains can now be obtained by computing the functions $K(\lambda)$, $\delta(\lambda)$, whose existence is guaranteed by the implicit function theorem. Since all poles can be placed, the function $\delta(\lambda)$ is not present and the remaining function can be obtained directly from the polynomial P(s) above. This gives

specified roots: $\lambda(s) = P(s) = s^4 + \lambda_4 s^3 + \lambda_3 s^2 + \lambda_2 s + \lambda_1$

unspecified roots:

$K_{23} = d_3 - \lambda_4$

$K_{12} = d_2 - \lambda_3 - K_{23} d_3$

$K_{13} \neq 0$ otherwise arbitrary

K_{22} arbitrary

$K_{11} = d_1 - \lambda_2 - K_{23} d_2 + K_{12} K_{23} - K_{13} K_{22}$

$K_{21} = (-\lambda_1 - K_{23} d_1 + K_{11} K_{23})/K_{13}$

The final compensated system has the structure shown in Figure 14. More elaborate computational examples are treated in Appendix G.

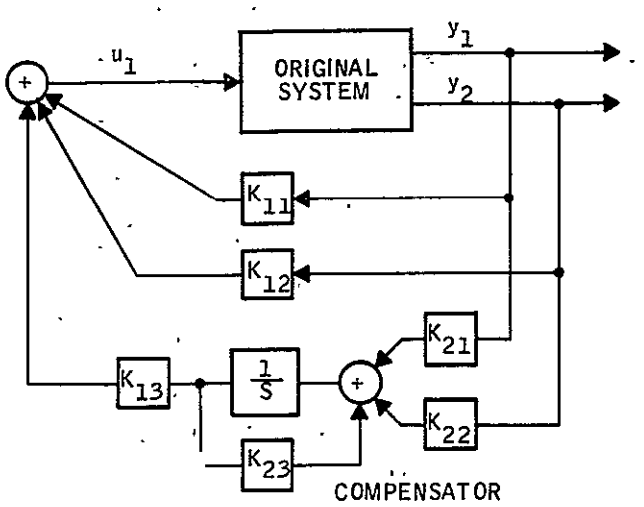


Figure 14. Example Compensator Design

A Pole-Placement Algorithm

In the example above, the system was sufficiently small to permit direct manual solution of the pole-placement problem. This will not be the case, of course, for large flexible-vehicle design problems. To establish computational feasibility for the proposed quality measures, therefore, it is necessary to develop computerized solutions able to handle high order systems. These are provided by the Newton Raphson algorithm shown in Figure 15.

This algorithm was used to carry out the trial computations in Appendix G, which deals with 6th- and 17th-order dynamic systems. Computation times were very reasonable for both problems. For the 17th-order case with two control inputs and four measurements, a single run consumes approximately 5 minutes on an SDS 9300 machine. About half of this time is devoted to the computation of generalized numerator coefficient vectors which present certain numerical challenges because of their large magnitudes for high-order problems. At present, a generalized eigenvalue routine is used to

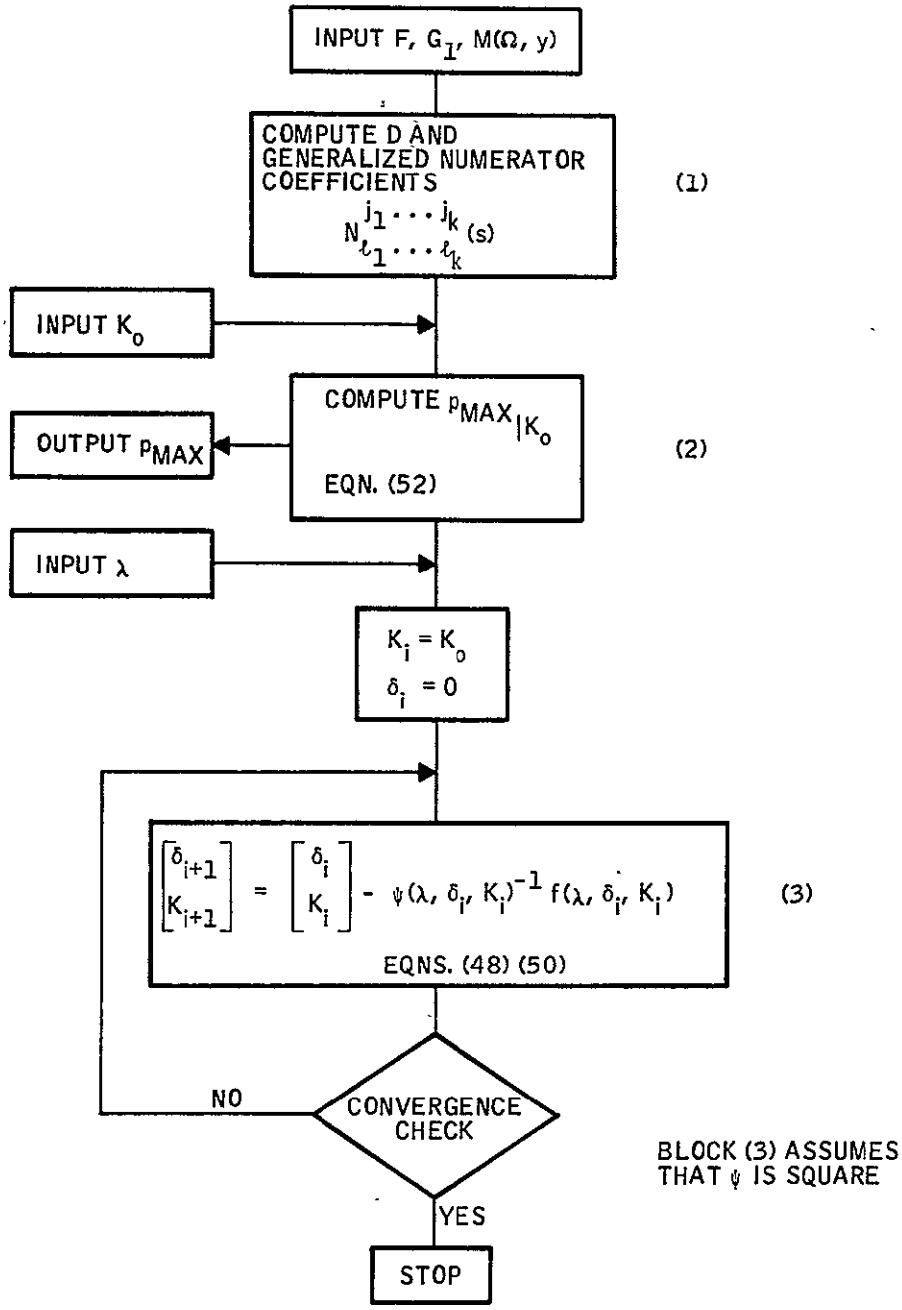


Figure 15. Pole-Placement Algorithm

perform the computations. This routine is discussed briefly in Appendix H, together with an alternate procedure which may prove more efficient in future applications of the pole-placement algorithm as an inner loop of the basic sensor choice algorithm of Figure 13. The alternate procedure utilizes the fact that higher-order numerators of equation (44) can be expressed in terms of first-order numerators (Ref. 18). This fact significantly reduces the computations required to evaluate the coefficient vectors for many sensor locations, y.

SECTION V
CONCLUSIONS

This study has produced partial answers to three fundamental problems in optimal controller design: (1) controller simplification, (2) sensitivity to model inaccuracy, and (3) sensor complement choice. Progress achieved on these problems are briefly summarized here along with unanswered questions.

In the area of controller simplification, a gradient iteration method and a compatible method for initialization were developed. With these methods a simplified controller was successfully designed for a rigid-body model of a launch vehicle. Parametric techniques were revised and used to design a simplified controller for a large flexible aircraft at one flight condition. The remaining question with respect to controller simplification is whether the techniques can be combined to handle flexible launch vehicle problems. There is no basic inconsistency between the methods, so it appears the question can be answered in the affirmative. However, verification can only come by actually exercising the techniques on a realistic problem.

The major result with regard to model inaccuracy is that a boundary point of the model's admissible parameters should be used to design an optimal insensitive controller. A necessary condition which such a boundary point must satisfy was derived. Computational aspects were considered and a numerical example corresponding to a flexible aircraft with unknown flexure frequencies was treated. The significant question remaining concerns admissible ranges of parameters. The magnitude of admissible parameter variations in the example was approximately 15 percent. At present such a figure must be determined experimentally on the computer for each individual problem. An a priori estimate is highly desirable.

To gain knowledge concerning the quality of sensor complements, the pole-placement capacity of a controller with a given sensor complement was studied. The following theoretical result was achieved for multi-input systems: to determine the maximum pole-placement capacity offered by a sensor complement, each input must be utilized, and indeed, the capacity of a given sensor complement is generally enhanced as more inputs are admitted. Furthermore, this capacity can be evaluated with reasonable computational loads for compensated as well as uncompensated systems. In fact, the minimal compensator required to yield complete pole-placement capability can be determined computationally. Algorithms were developed and exercised on up to 17th-order examples. The question of how pole-placement quality measures for a sensor complement relate to measures of quality with respect to controller performance has been left unanswered.

The positive results achieved in each of the three areas of investigation make it possible to apply the stochastic constrained-response technology to the design of a controller for mated ascent of the Space Shuttle Vehicle.

REFERENCES

1. Edinger, L. D., et al, "Design of a Load-Relief Control System," Final Technical Report, Honeywell Document 12013-FR1, May 1966, NASA Contract Report CR-61169, April 21, 1967.

2. Bender, M. A. (Honeywell), Burris, P. M. (Boeing), "Aircraft Mode Alleviation and Mode Stabilization (LAMS)" Final Report, AFFDL-TR-68-158, April 1969.

3. Harvey, C. A., "Application of Optimal Control Theory to Launch Vehicles," Final Technical Report, Honeywell Document 12073-FR1 and 2, July 1968.

4. Axsäter, Sven, "Sub-Optimal Time-Variable Feedback Control of Linear Dynamic Systems with Random Inputs," Int. J. Control, 1966, 4, No. 6, pp. 549-566.

5. Johnson, T. L., "The Aerodynamic Surface Location Problem in Optimal Control of Flexible Aircraft," Report ESL-R-387, MIT, June 1969.

6. Stein, G., Henke, A. H., "A Design Procedure and Handling Quality Criteria for Lateral-Directional Flight Control Systems," Air Force Flight Dynamics Laboratory Technical Report No. AFFDL-TR-70-152, Wright Patterson Air Force Base, Ohio, February 1971.

7. Scharmack, D. K., "An Initial Value Method for Trajectory Optimization" Advances in Control Systems, 5, L. T. Leondes, Editor, Academic Press, New York, 1967.

8. Courant, Differential and Integral Calculus, Interscience, New York, 1947.

9. VanDierendonk, A. J., "Design Method for Fully Augmented Systems for Variable Flight Conditions," Honeywell ADG Report 12260-CRS 6, April 1971.

10. "Aircraft Load Alleviation and Mode Stabilization (LAMS)", C-5A System Analysis and Synthesis, AFFDL-TR-68-162, November 1969.

11. Skelton, G. B., "Sensor Location," Honeywell Interoffice Correspondence MR 19669, April 30, 1969.

12. Wonham, W. M., "On Pole Assignment in Multi-Input Controllable Linear Systems," IEEE Trans. Auto. Control, AC-12, No. 6, pp. 660-665.

13. Simon, J. D. and S. K. Mitter, "A Theory of Modal Control," Inform. and Control, 13, pp. 316-353, October 1968.

14. Jameson, A., "Design of Single Input Systems with Specified Roots Using Output Feedback," IEEE Trans. Auto. Control, AC-15, pp. 345-347, June 1970.

15. Davison, E. J., "On Pole Assignment in Linear Systems with Incomplete State Feedback," IEEE Trans. Auto. Control, AC-15, pp. 348-350, June 1970.

16. Brash, F. M., Jr., and J. B. Pearson, "Pole Placement Using Dynamic Compensators," IEEE Trans. Auto. Control, AC-15, No. 1, pp. 34-43.

17. Pearson, J. B., and C. Y. Ding, "Compensator Design for Multivariable Linear Systems," IEEE Trans., AC-14, No. 2, April 1969, pp. 130-134.

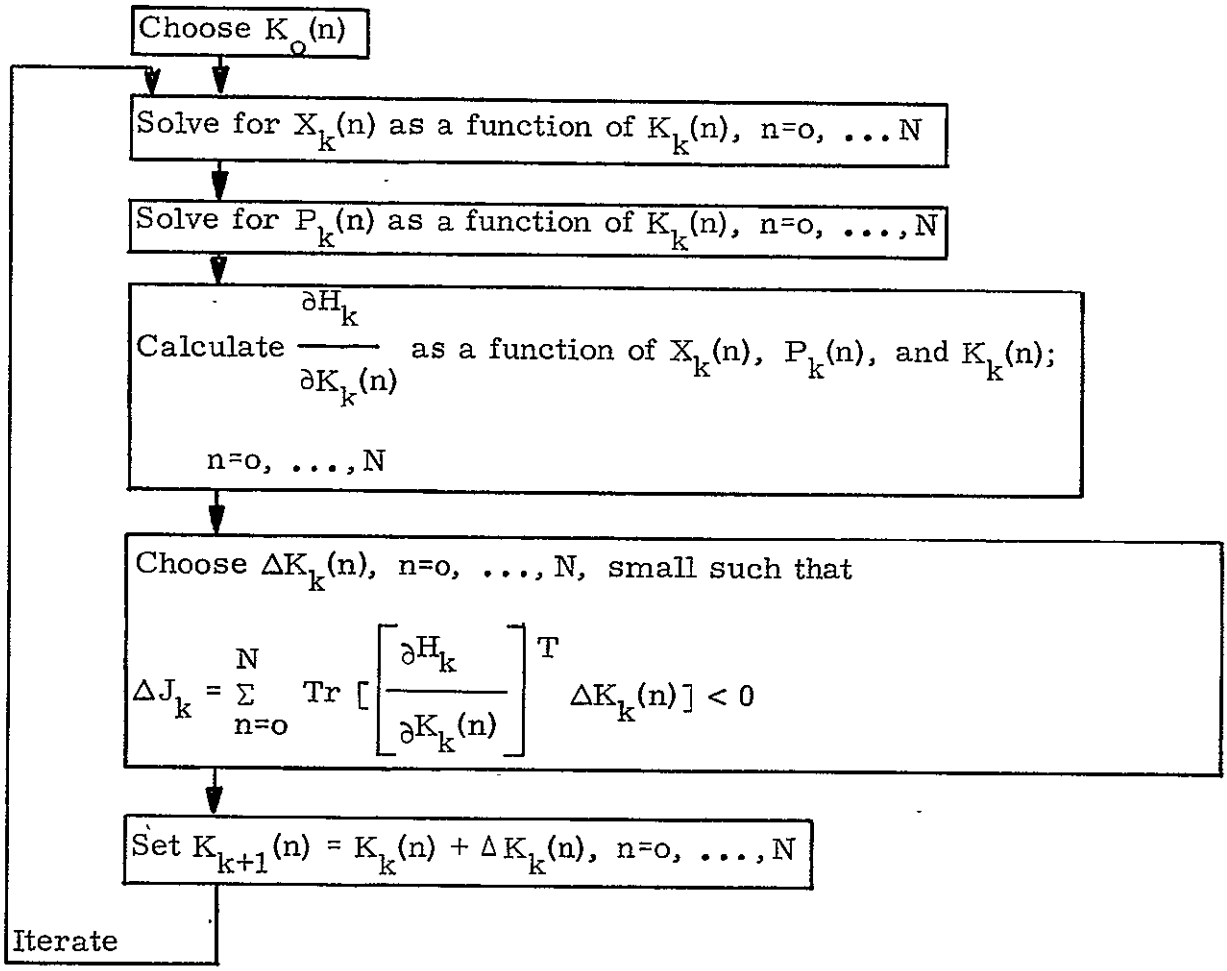
18. McRuer, D. T., I. L. Ashkenas, and H. R. Pass, "Analysis of Multiloop Vehicular Control Systems, ASD-TDR-62-1014, March 1964.

19. Peters, G. and J. H. Wilkinson, "$Ax = \lambda Bx$ and the Generalized Eigenproblem," SIAM J. Numerical Analysis, 7, No. 4, Dec. 1970, pp. 479-492.

20. Salmon, D. M., "Minimax Controller Design," IEEE Trans. Auto. Control, AC-13, No. 4, August 1968, pp. 369-376.

APPENDIX A
GRADIENT SEARCH FORMULATION

The following flow chart describes the gradient method used for optimizing fixed-form controllers for time-varying systems. In the chart, $K_k(n)$ are the feedback gains of the k^{th} iteration for time n, H_k is the Hamiltonian of the k^{th} iteration, $P_k(n)$ is the co-state matrix of the k^{th} iteration for time n, and $X_k(n)$ is the covariance matrix of the k^{th} iteration for time n.



If $J = J^{**}$ (the quadratic cost), the Hamiltonian H_k is

$$H_k = TR\,[Q(N)H_1(N)X_k(N)H_1^T(N)]$$

$$+ TR\left[\sum_{n=o}^{N-1} \Delta t\, Q(n)\,(H_1(n)+D_1(n)K_k(n)M(n))X_k(n)\,(H_1(n)+D_1(n)K_k(n)M(n))^T\right]$$

$$+ TR\left[\sum_{n=o}^{N-1} P_k(n+1)(X(n+1)-X(n))\right]$$

where $Q(n)$ is the quadratic weighting matrix for time n, Δt is the sampling interval, $M(n)$ is the measurement matrix for time n, and $H_1(n)$, $D_1(n)$ are the usual system parameters.

To solve for $X_k(n)$, use the following sample data covariance solution (forward integration):

$$X_k(n+1) = [A(n)+B_1K_k(n)M(n)]X_k(n)\,[A(n)+B_1K_k(n)M(n)]^T$$

$$+(\Delta t)^{-1}B_3(n)W_1(n)B_3^T(n)$$

where $A(n)$, B_1, $B_3(n)$, and $W_1(n)$ are the usual system parameters. $X_k(0)$ is known and is constant.

To solve for the co-state matrices $P_k(n)$, set

$$\frac{\partial H_k}{\partial X_k(n)} = \frac{P_k(n+1) - P_k(n)}{\Delta t}$$

which has the solution (backward integration) with

$$P_k(N) = H_1(N)^T Q(N) H_1(N):$$

$$P_k(n) = [A(n)+B_1 K_k(n) M(n)]^T P_k(n+1)[A(n)+B_1 K_k(n) M(n)] +$$

$$\Delta t[H_1(n)+D_1(n)K_k(n)M(n)]^T Q(n)[H_1(n)+D_1(n)K_k(n)M(n)]$$

These equations yield the solution

$$\frac{\partial H_k}{\partial K_k(n)} = 2B_1{}^T P_k(n+1)[A(n)+B_1(n)K_k(n)M(n)]X_k(n)M^T(n)$$

$$+2\Delta t\, D_1{}^T(n)Q(n)\,[H_1(n)+D_1(n)K_k(n)M(n)]X_k(n)M^T(n)$$

with

$$\frac{\partial H_k}{\partial K_k(N)} = 0.$$

To choose $\Delta K_k(n)$ small such that

$$\Delta J_k = \sum_{n=0}^{N} Tr\,[\frac{\partial H_k{}^T}{\partial K_k(n)} \Delta K_k(n)] \leq 0,$$

we choose

$$Tr\,[\frac{\partial H_k{}^T}{\partial K_k(n)} \Delta K_k(n)] \leq 0$$

for all n, which is true if

$$\frac{\partial H_k}{\partial K_{k_{ij}}(n)} \Delta K_{k_{ij}}(n) \leq 0$$

for each i, j and n, which is true if $\frac{\partial H_k}{\partial K_{k_{ij}}(n)}$ and $\Delta K_{k_{ij}}(n)$ are of opposite sign.

This was accomplished by setting (for $\epsilon > 0$)

$$\Delta K_{k_{ij}}(n) = -\epsilon \left| K_{k_{ij}}(n) \right| \frac{\partial H_k}{\partial K_{k_{ij}}(n)} \Bigg/ \sqrt{\sum_{i,\underline{j},n} \left(\frac{\partial H_k}{\partial K_{k_{ij}}(n)} \right)^2} \quad \dagger$$

If it is desired that J* (the upper bound on the probability of mission failure) is minimized instead of J**, simply set all the Q(n) of the above equations to

$$Q(n) = Q_k(n) = \frac{\partial J^*}{\partial S_k(n)}$$

where

$$S_k(n) = [H_1(n) + D_1(n) K_k(n) M(n)] X_k(n) [H_1(n) + D_1(n) K_k(n) M(n)]^T$$

---

†For the launch vehicle, we have a single input; this means i=1. Normalizing over each gain individually consists in removal of summation over j in radicand.

This is true because

$$\frac{\partial J^*}{\partial K_{k_{ij}}(n)} = \mathrm{Tr}\left[\frac{\partial J^*}{\partial S_k(n)} \cdot \frac{\partial S_k(n)}{\partial K_{k_{ij}}(n)}\right]$$

where

$$\frac{\partial J^{**}}{\partial K_{k_{ij}}(n)} = \mathrm{Tr}\left[Q(n)\frac{\partial S_k(n)}{\partial K_{k_{ij}}(n)}\right]$$

which are identical if

$$Q(n) = Q_k(n) = \frac{\partial J^*}{\partial S_k(n)}$$

APPENDIX B
SIMPLIFICATION OF TIME-VARYING GAINS

A method is proposed for further simplification of time-varying optimal controllers through time linearization. It is proposed that the gain matrix K(t) be constrained to be piecewise linear with respect to time. That is, let

$$K(t) = a_\ell \cdot (t-t_\ell) + b_\ell \text{ for } t_{\ell-1} \le t \le t_\ell, \quad \ell = 1, \ldots, p \tag{B1}$$

where $t_o = 0$ and $t_p = T$.

The method considers optimizing via a gradient scheme with respect to the a_ℓ and b_ℓ with a constraint for continuity of K(t). The breakpoints t_ℓ are determined visually from practicalized gains obtained prior to these parameter optimizations.

Also discussed is the determination of the deterministic input for the simplified controller.

FORMULATION OF THE PROBLEM

Figures B1 through B5 show how five time-varying gains may be split up in a piecewise time sense. It seems that this could be done reasonably with six breakpoints, t_o through t_6. Actually, the set of breakpoints would be the union of the necessary breakpoints necessary for each individual time-varying gain. The knowledge of the physics involved along with the visual observation of the gains would determine where the breakpoints should be. From a practical point of view it is advantageous to have as few breakpoints as possible.

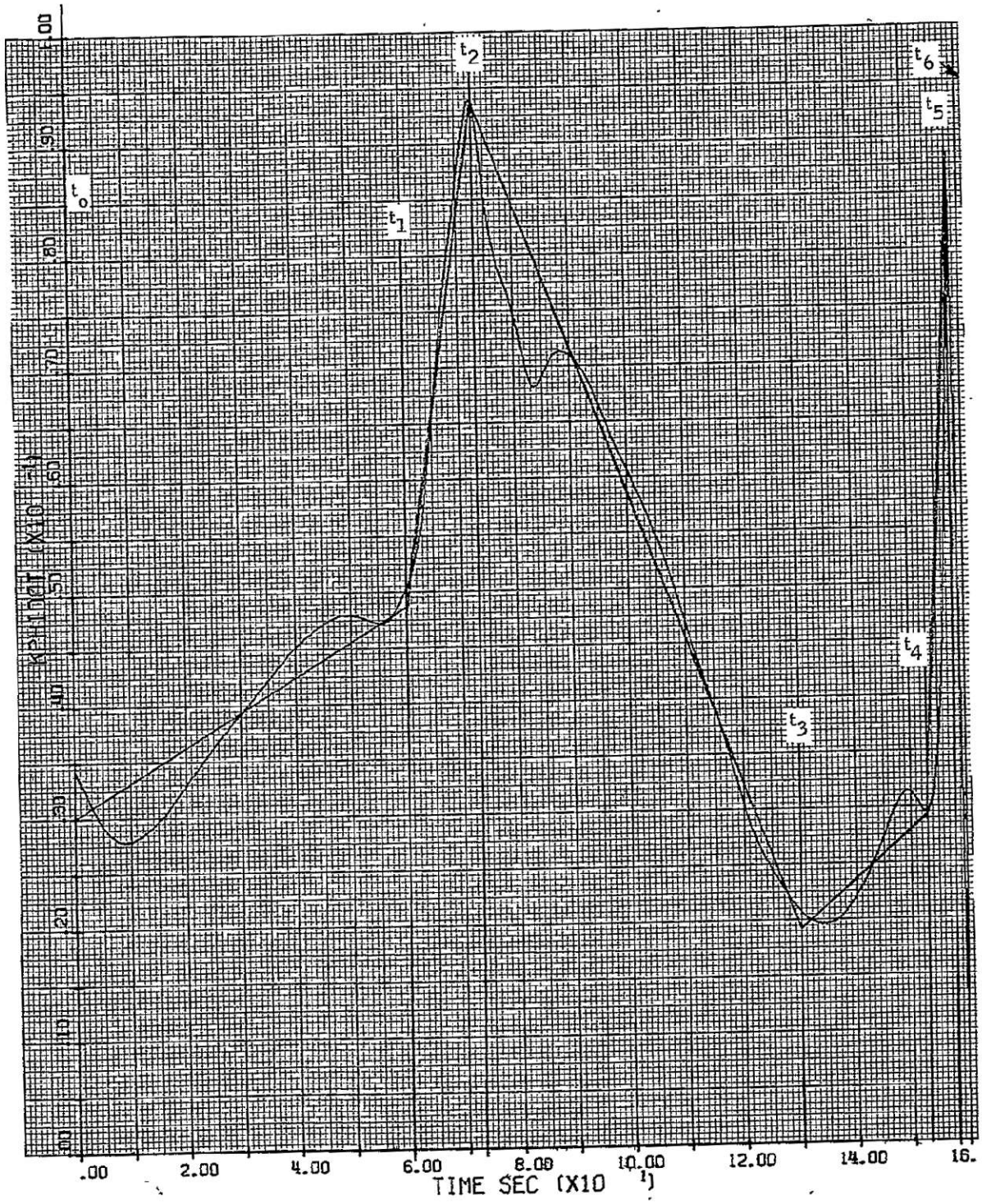


Figure B1. Optimal $\dot{\phi}$ Gain with Initial Approximation

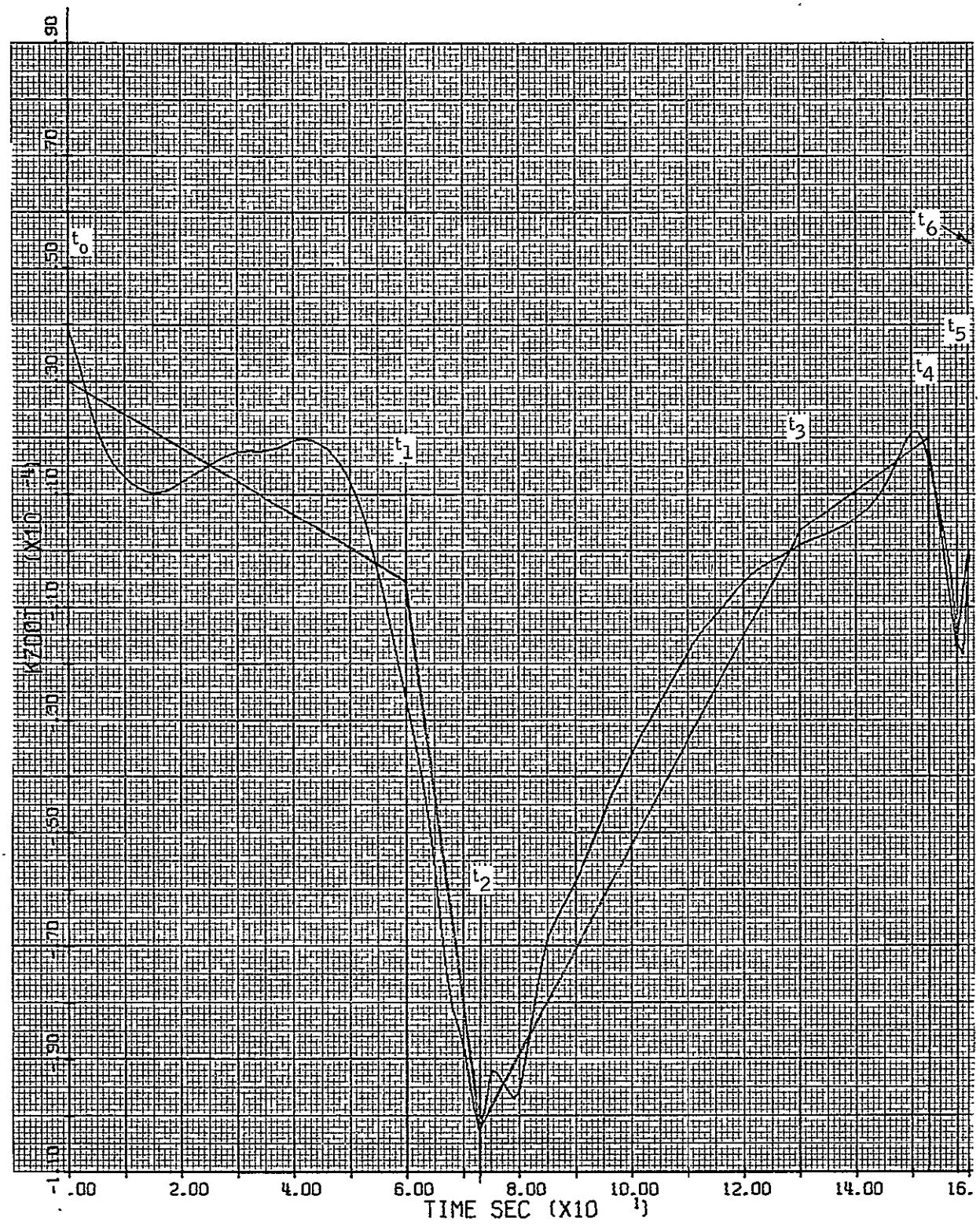


Figure B2. Optimal $\dot{z}$ Gain with Initial Approximation

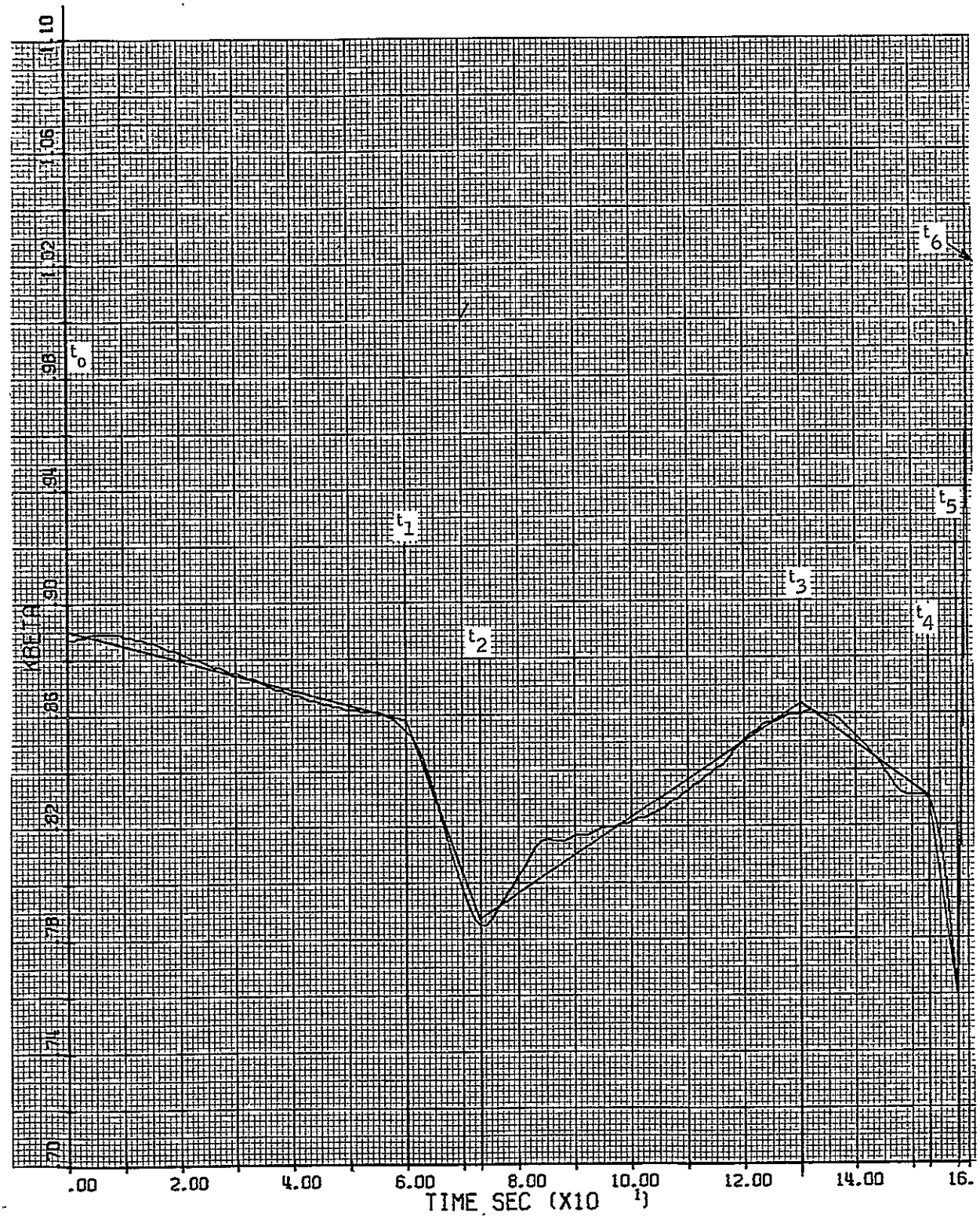


Figure B3. Optimal β Gain with Initial Approximation

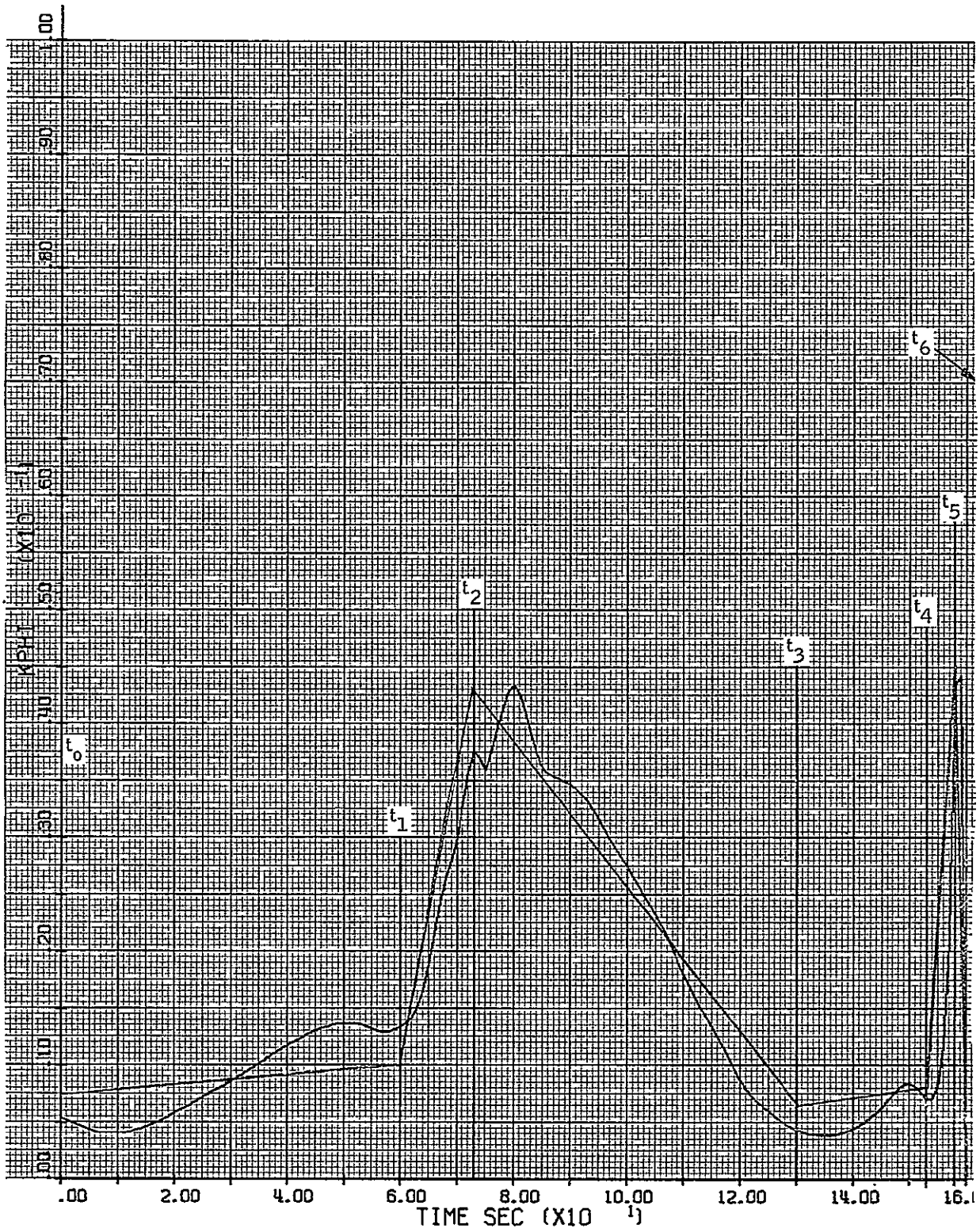


Figure B4. Optimal ϕ Gain with Initial Approximation

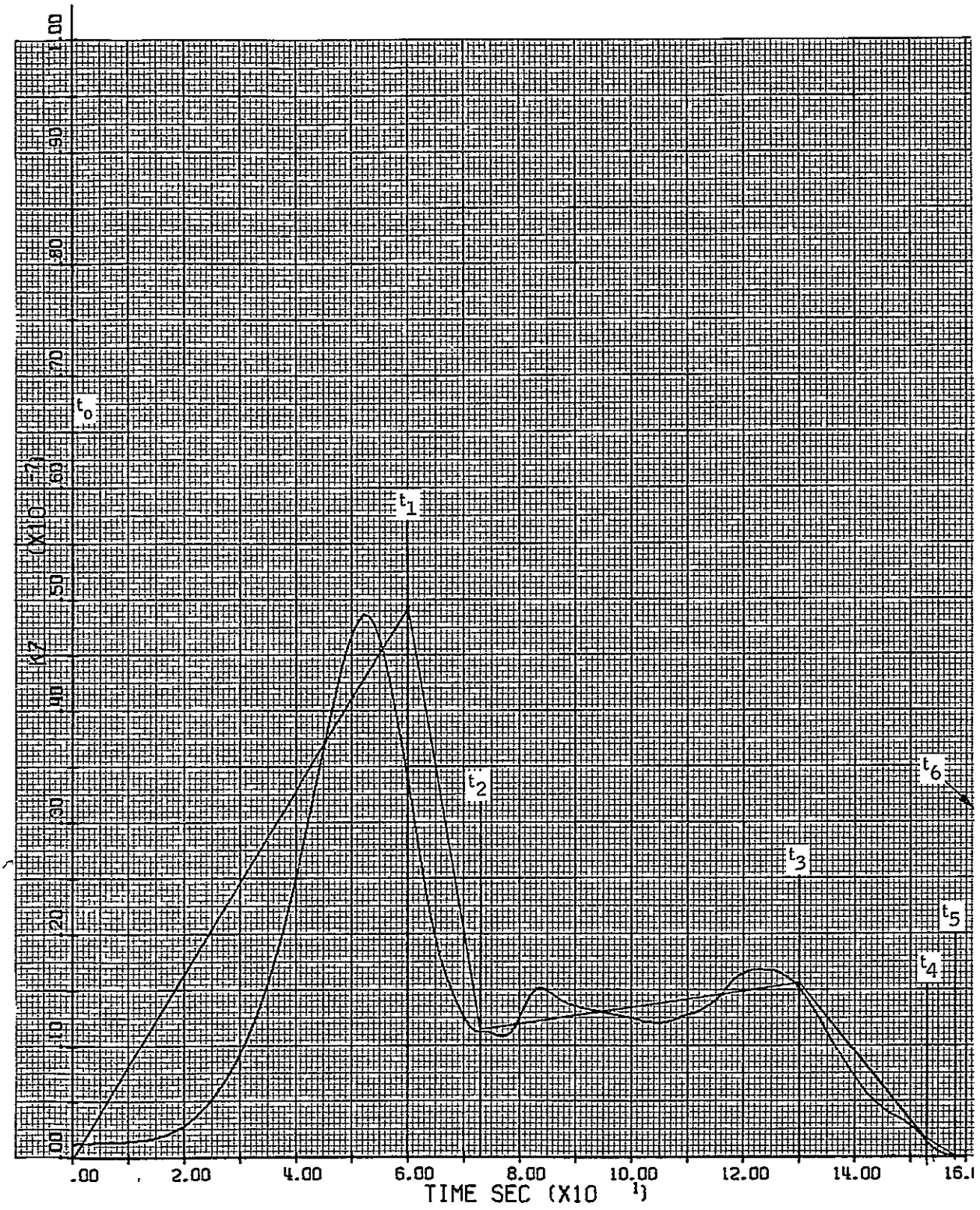


Figure B5. Optimal z Gain with Initial Approximation

Given the set of breakpoints, we then define an initial set of gains (in difference equation formulation)

$$K_o(n) = a_{\ell_o}(n-n_\ell) + b_{\ell_o}\ ;\ n_{\ell-1} \le n \le n_\ell,\ \ell=1, \ldots, p \qquad \text{(B2)}$$

where

$$b_{p_o} = K^*(N)\dagger \text{ (Practicalized final time gain)} \qquad \text{(B3)}$$

$$b_{\ell_o} = a_{\ell+1_o}(n_\ell - n_{\ell+1}) + b_{\ell+1_o}\ ;\ \ell=1, \ldots, p-1 \qquad \text{(B4)}$$

and the a_{ℓ_o} are the slopes of the lines determined graphically or through parameter optimization† where in conjunction with equations (B3) and (B4)

$$a_{\ell_o} = \left\{ \sum_{n=n_{\ell-1}}^{n_\ell} \left[b_{\ell_o} - K^*(n) \right] (n-n_\ell) \right\} \Bigg/ \sum_{n=n_{\ell-1}}^{n_\ell} (n-n_\ell)^2 \qquad \text{(B5)}$$

---

†b_{p_o} may also be determined through parameter optimization by solving the following two equations simultaneously:

$$a_{p_o} = \left\{ \sum_{n=n_{p-1}}^{N} \left[b_{p_o} - K^*(n) \right] (n-N) \right\} \Bigg/ \sum_{n=n_{p-1}}^{N} (n-N)^2$$

$$b_{p_o} = \left\{ \sum_{n=n_{p-1}}^{N} \left[K^*(n) - a_{p_o}(n-N) \right] \right\} \Big/ (N-n_{p-1})$$

in which case $\sum_{n=n_{p-1}}^{N} [K^*(n) - K_o(n)]^2$ is minimized with respect to a_{p_o} and b_{p_o}.

where $K^*(n)$ are the set of practilized time-varying gains. Then $\sum_{n=n_{\ell-1}}^{n_\ell} [K^*(n) - K_o(n)]^2$ is minimized with respect to a_{ℓ_o}. Time n_p is equal to N; n_o is equal to zero.

When optimizing on the a_ℓ and b_ℓ with the constraint for continuity of K(t), the Hamiltonian is

$$H_k = TR[Q(N)H_1(N)X_k(N)H_1^T(N)]$$
$$+ TR\left\{\sum_{n=o}^{N-1} \Delta t Q(n)[H_1(n) + D_1(n)K_k(n)M(n)]X_k(n)[H_1(n) + D_1(n)K_k(n)M(n)]^T\right\} + TR\left\{\sum_{n=o}^{N-1} P_k(n+1)[X(n+1) - X(n)]\right\} \quad \text{(B6)}$$

where Q(n) is the quadratic weighting matrix for time n, Δt is the sampling interval, $P_k(n)$ is the costate matrix and $X_k(n)$ is the covariance matrix of the k^{th} iteration for time n, M(n) is the measurement matrix for time n, and $H_1(n)$ and $D_1(n)$ are the usual system parameters. The $K_k(n)$ are defined as

$$K_k(n) = a_{\ell_k}(n-n_\ell) + b_{\ell_k}; \; n_{\ell-1} \le n \le n_\ell, \; \ell=1, \ldots, p \quad \text{(B7)}$$

where

$$b_{p_k} = K^*(N)\dagger \quad \text{(B8)}$$

with the constraint that

$$b_{\ell_k} = a_{\ell+1_k}(n_\ell - n_{\ell+1}) + b_{\ell+1_k}; \; \ell=1, \ldots, p-1 \quad \text{(B9)}$$

†See footnote on page B11.

which can be written as

$$b_{\ell_k} = \sum_{j=\ell+1}^{p} \left[a_{j_k} (n_{j-1} - n_j) \right] + b_{p_k} \tag{B10}$$

From equation (B10), we see that the b_{ℓ_k} are functions of the a_{j_k}, where j is greater than ℓ, and b_{p_k}, and not the a_{ℓ_k}.

To solve for $X_k(n)$, use the following sample data covariance solution:

$$\begin{aligned} X_k(n+1) = {} & [A(n) + B_1 K_k(n) M(n)] X_k(n) [A(n) + B_1 K_k(n) M(n)]^T \\ & + (\Delta t)^{-1} B_3(n) W_1(n) B_3{}^T(n) \end{aligned} \tag{B11}$$

where

$A(n)$, B_1 $B_3(n)$, and $W_1(n)$ are the usual system parameters.

$X_k(0)$ is known and is constant.

To solve for the costate matrices $P_k(n)$, set

$$\frac{\partial H_k}{\partial X_k(n)} = - \frac{P_k(n+1) - P_k(n)}{\Delta t} \tag{B12}$$

which has the solution with $P_k(N) = H_1(N)^T Q(N) H_1(N)$

$$\begin{aligned} P_k(n) = {} & [A(n) + B_1 K_k(n) M(n)]^T P_k(n+1) [A(n) + B_1 K_k(n) M(n)] \\ & + \Delta t [H_1(n) + D_1(n) K_k(n) M(n)]^T Q(n) [H_1(n) + D_1(n) K_k(n) M(n)] \end{aligned} \tag{B13}$$

These equations yield the solution

$$\frac{dH_k}{da_{\ell_k}} = \sum_{n=o}^{N} \frac{\partial H_k}{\partial K_k(n)} \cdot \frac{dK_k(n)}{da_{\ell_k}} \tag{B14}$$

$$\frac{dK(n)}{da_\ell} = \begin{cases} 0 & n_\ell \le n \\ n - n_\ell & n_{\ell-1} \le n \le n_\ell \\ n_{\ell-1} - n_\ell & n \le n_{\ell-1} \end{cases}$$

where

$$\frac{\partial H_k}{\partial K_k(n)} = 2B_1^T P_k(n+1) \left[A(n) + B_1(n)\, K_k(n) M(n)\right] \bar{X}_k(n)\, M^T(n)$$
$$+2\Delta t\, D_1^T(n)\, Q(n) \left[H_1(n) + D_1(n)\, K_k(n) M(n)\right] \bar{X}_k(n)\, M^T(n) \tag{B15}$$

Note that

$$\frac{dK(n)}{da_\ell} = \frac{db_j}{da_\ell} = n_{\ell-1} - n_\ell \quad \text{for } n_{j-1} \le n \le n_j,\ j < \ell.$$

We can use equation (B15) computed backward in time to choose $\Delta K_k(n)$ in a gradient search, where

$$\Delta K_k(n) = \Delta a_{\ell_k}(n - n_\ell) + \Delta b_{\ell_k}; \quad n_{\ell-1} \le n \le n_\ell, \quad \ell = 1, \ldots, p \tag{B16}$$

where

$$\Delta a_{\ell_{ki}} = -\epsilon \frac{\left|a_{\ell_{ki}}\right|}{\sqrt{\sum_{\ell=1,p} a_{\ell_k} a_{\ell_k}^T}} \cdot \frac{\partial H_k}{\partial a_{\ell_{ki}}} \tag{B17}$$

The subscript i represents the i^{th} component and ϵ is the maximum percentage change in the a_{ℓ_k}.

Also

$$\Delta b_{\ell_k} = b_{\ell_k} - b_{\ell_{k-1}}$$

$$= \sum_{j=\ell+1}^{P} \Delta a_{j_k} (n_{j-1} - n_j) + (b_{p_k} - b_{p_{k-1}}) \tag{B18}$$

where

$$b_{p_k} = K^*(N). \dagger$$

The gradient search is set up as in Appendix A. If it is desired to minimize a nonquadratic cost J^*, let

$$Q(n) = Q_k(n) = \frac{\partial J^*}{\partial S_k(n)} \tag{B19}$$

---

†This b_{p_k} may also be a variable, where

$$\frac{\partial H_k}{\partial b_{p_k}} = \sum_{n=o}^{N} \frac{\partial H_k}{\partial K_k(n)} \quad \frac{dK_k(n)}{db_{pk}} = \sum_{n=o}^{N} \frac{\partial H_k}{\partial K_k(n)}$$

instead of being fixed since this parameter is not constrained.

Note

$$\frac{dK_k(n)}{db_{p_k}} = 1 \text{ for all n from (B7) and (B10)}$$

where

$$S_k(n) = [H_1(n) + D_1(n)\, K_k(n) M(n)]\, X_k(n)\, [H_1(n) + D_1(n)\, K_k(n) M(n)]^T$$

as discussed in Appendix A.

THE DETERMINISTIC INPUT

From the perfect sensing gains $\tilde{K}(n)$, the deterministic input $\tilde{f}(n)$ is defined. The deterministic controller $\bar{u}$ is then

$$\bar{u}(n) = \tilde{K}(n)\bar{x}(n) + \tilde{f}_n \tag{B20}$$

Since this is the optimum deterministic controller, it may as well be used with the simplified controller gains $K(n)$, and a new deterministic input $f^*(n)$, where

$$f^*(n) = \bar{u}(n) - K(n)M(n)\bar{x}(n) \tag{B21}$$

where $\bar{u}(n)$ is the controller defined in equation (B20) and $\bar{x}(n)$ is the mean state vector found by

$$\bar{x}(n+1) = A(n)\bar{x}(n) + B_1\bar{u}(n) + B_2(u)\bar{v}_w(n) \tag{B22}$$

which is the same as the optimum since $\bar{u}(n)$ is optimum.

In order to further simplify the deterministic input to the form

$$f(n) = c_\ell \cdot (n - n_\ell) + d_\ell \quad \text{for} \quad n_{\ell-1} \le n \le n_\ell \; ; \quad \ell = 1, \ldots, q \tag{B23}$$

an initial set of c_ℓ and d_ℓ can be found with a straight parameter optimization where

$$d_{q_o} = \left\{ \sum_{n=n_{q-1}}^{N} \left[f^*(n) - c_{q_o}(n-N) \right] \right\} \Big/ (N-n_{q-1})$$

$$c_{\ell_o} = \left\{ \sum_{n=n_{\ell-1}}^{n_\ell} \left[d_{\ell_o} - f^*(n) \right] \quad (n-n_\ell) \right\} \Big/ \sum_{n=n_{\ell-1}}^{n_\ell} (n-n_\ell)^2; \quad \ell=1, \ldots, q \tag{B24}$$

and

$$d_{\ell_o} = c_{\ell+1_o}(n_\ell - n_{\ell+1}) + d_{\ell+1_o}; \quad \ell=1, \ldots, q-1 \tag{B25}$$

and then use a gradient method to improve on these. It is obvious from Figure B6, that the same breakpoints as for the K(n) cannot be used. For the gradient method we would use the same procedure as for the K(n) except that the gradient is now for the k^{th} iteration

$$\frac{\partial \bar{H}}{\partial c_{\ell_k}} = \sum_{n=n_{\ell-1}}^{n_\ell} \frac{\partial \bar{H}}{\partial f_k(n)}$$

$$= \sum_{n=o}^{N} \frac{\partial \bar{H}}{\partial f_k(n)} \cdot \frac{\partial f_k(n)}{\partial c_{\ell_k}}$$

where

$$\frac{\partial f_k(n)}{\partial c_{\ell_k}} = \begin{cases} 0 & n_\ell \le n \\ n-n_\ell & n_{\ell-1} \le n \le n_\ell \\ n_{\ell-1}-n_\ell & n \le n_{\ell-1} \end{cases}$$

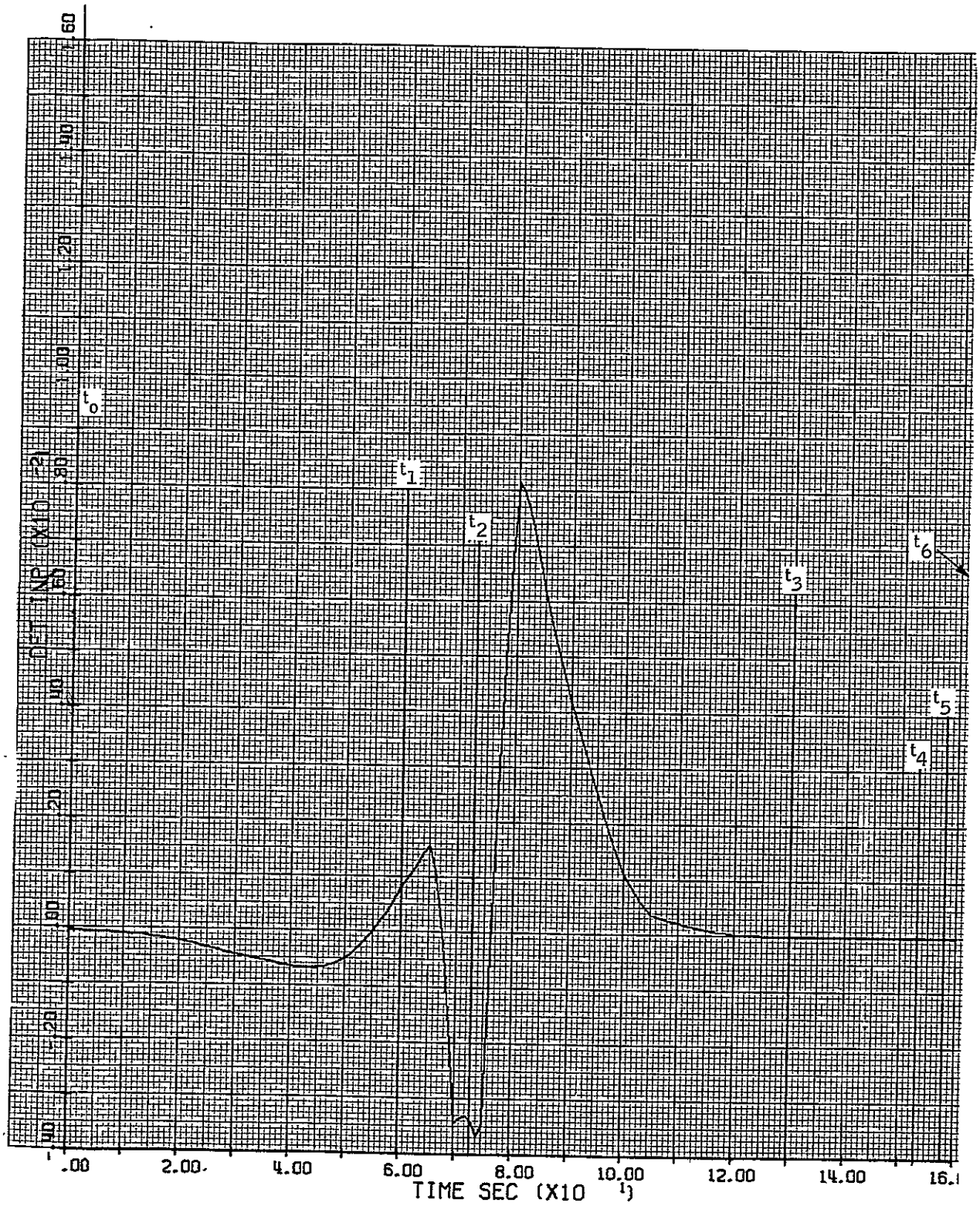


Figure B6. Optimal Deterministic Input

and

$$\frac{\partial \bar{H}}{\partial d_{q_k}} = \sum_{n=o}^{N} \frac{\partial \bar{H}}{\partial f_k(n)} \cdot \frac{\partial f_k(n)}{\partial d_{q_k}} = \sum_{n=o}^{N} \frac{\partial \bar{H}_k}{\partial f_k(n)} \quad \text{since}$$

$$\frac{\partial f_k(n)}{\partial d_{q_k}} = 1 \text{ for all n.}$$

where $\bar{H}$ is the Hamiltonian associated with the mean responses and

$$\frac{\partial \bar{H}}{\partial f_k(n)} = \frac{B_1^T}{\Delta t} \lambda_k(n+1) + 2D_1^T(n)Q(n) \left\{[H_1(n) + D_1(n)\, K(n)M(n)]\, \bar{x}_k(n) + D_1(n)\, f_k(n) + D_2(n)\, \bar{v}_\omega(n)\right\}$$

where,

$$\lambda_k(N) = 2H_1(N)^T\, Q(N)\, [H_1(N)\, \bar{x}_k(N) + D_2(N)\, \bar{v}_\omega(N)]$$

$$\lambda_k(n) = [A(n) + B_1\, K(n)M(n)]^T\, \lambda_k(n+1) + 2\Delta t[H_1(n) + D_1(n)\, K(n)M(n)]^T\, Q(n) \left\{[H_1(n) + D_1(n)\, K(n)M(n)]\, \bar{x}_k(n) + D_1(n)\, f_k(n) + D_2(n)\, \bar{v}_\omega(n)\right\}$$

and

$$\bar{x}_k(n+1) = [A(n) + B_1\, K(n)M(n)]\, \bar{x}_k(n) + B_1\, f_k(n) + B_2(n)\, \bar{v}_\omega(n)$$

with $\bar{x}_k(0) = \bar{x}_o$.

The procedure is then to compute $x_k(n)$ forward, and $\lambda_k(n)$ and the gradient backwards, compute $c_{\ell_{k+1}}$ and $d_{q_{k+1}}$ by

$$c_{\ell_{k+1}} = c_{\ell_k} + \Delta c_{\ell_k}$$

and

$$d_{q_{k+1}} = d_{q_k} + \Delta d_{q_k}$$

or compute $f_{k+1}(n)$ by

$$f_{k+1}(n) = f_k(n) + \Delta f_k(n)$$

where

$$\Delta f_k(n) = \Delta c_{\ell_k} \cdot (n - n_{\ell}) + \Delta d_{\ell_k}; \quad n_{\ell-1} \leq n \leq n_{\ell}, \quad \ell = 1, \ldots, q$$

The new $f_{k+1}(n)$ are used to repeat the procedure for the next step.

APPENDIX C
TIME-VARYING GAINS FROM GRADIENT ITERATIONS

This appendix consists of graphs of the gains found in Section II, Iteration Method for Time-Varying Systems. They are presented as the following figures:

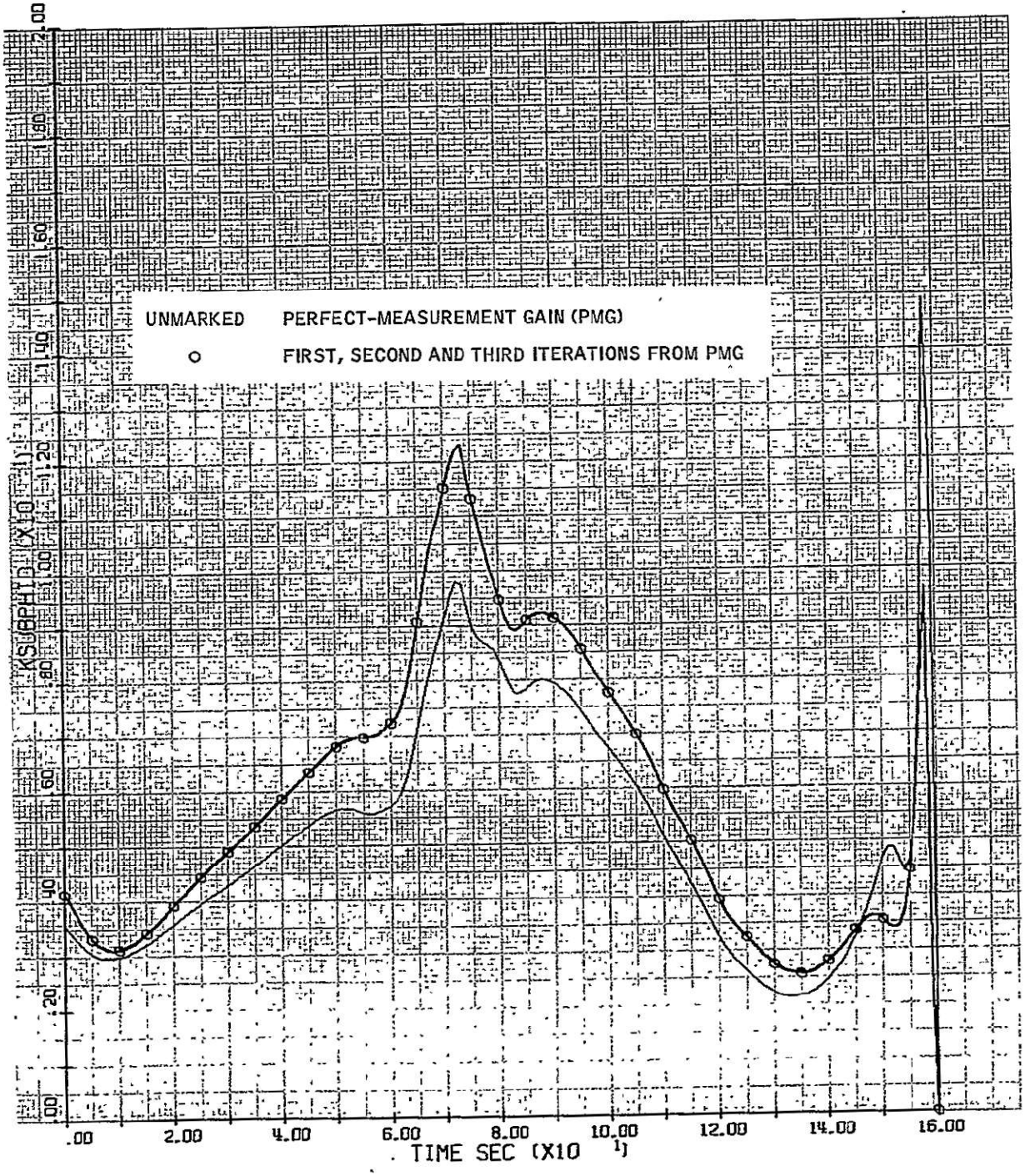


Figure C1. $\dot{\phi}$ Gain, Iterations from Perfect-Measurement Gain

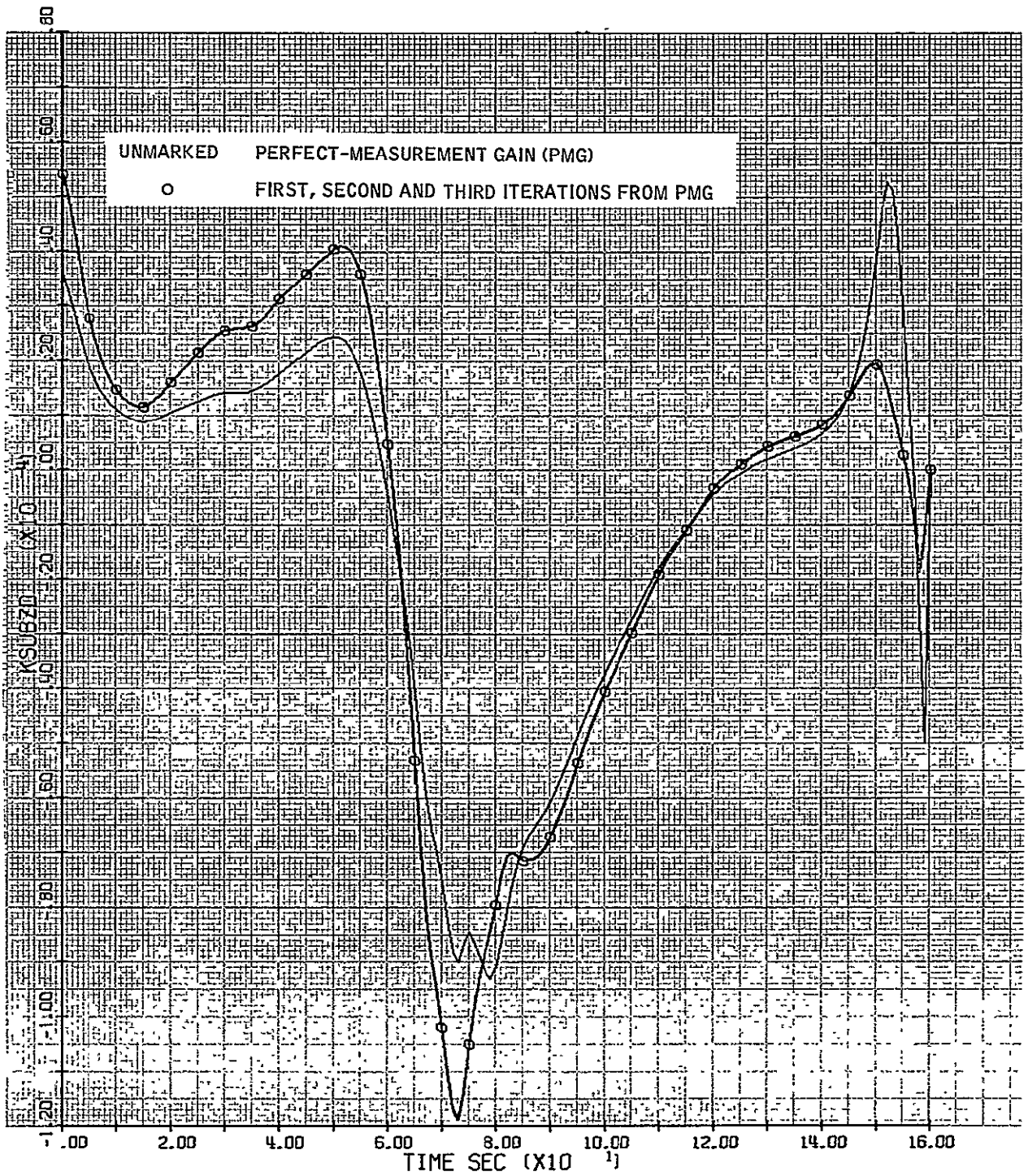


Figure C2. $\dot{z}$ Gain, Iterations from Perfect-Measurement Gain

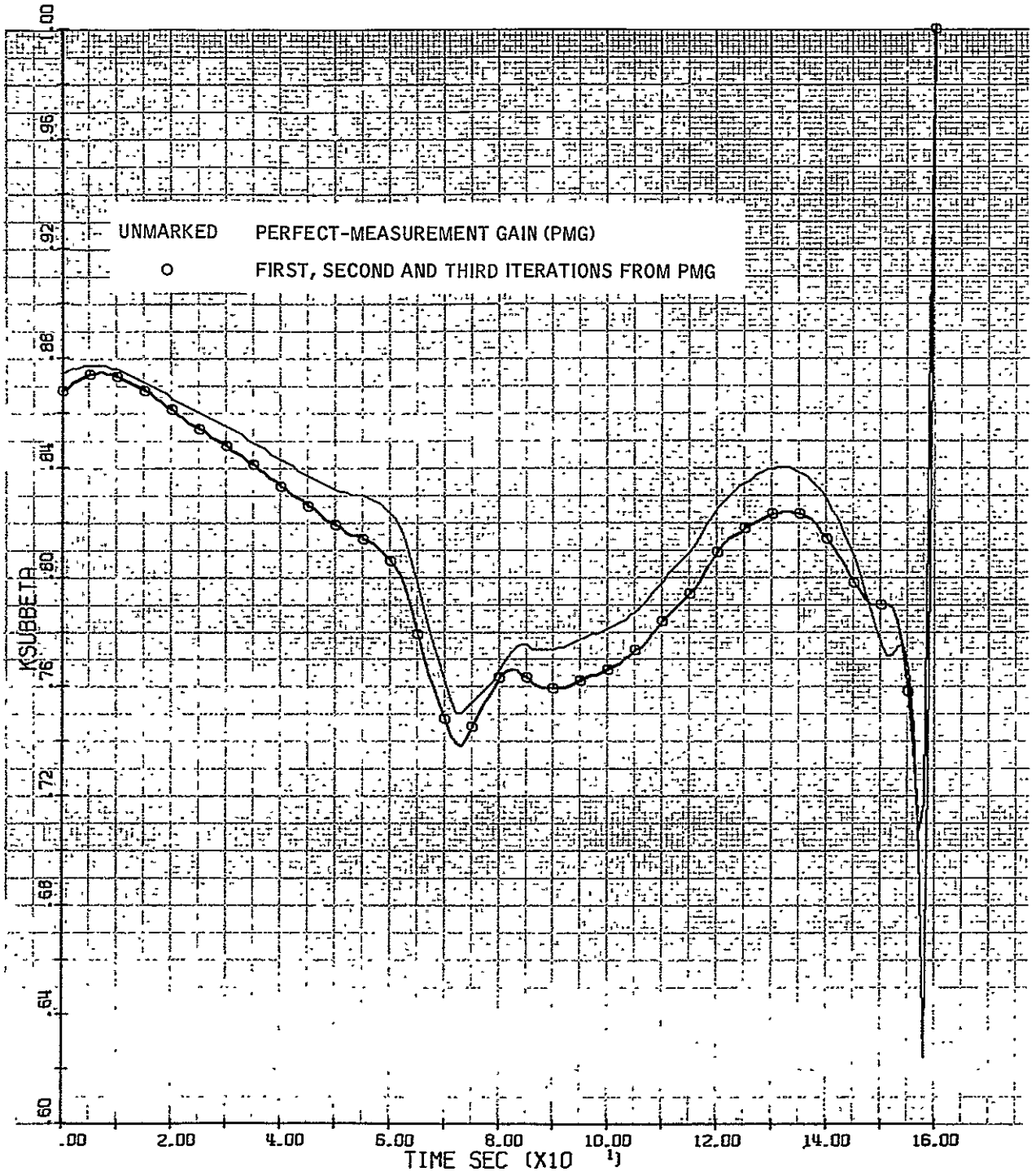


Figure C3. β Gain, Iterations from Perfect-Measurement Gain

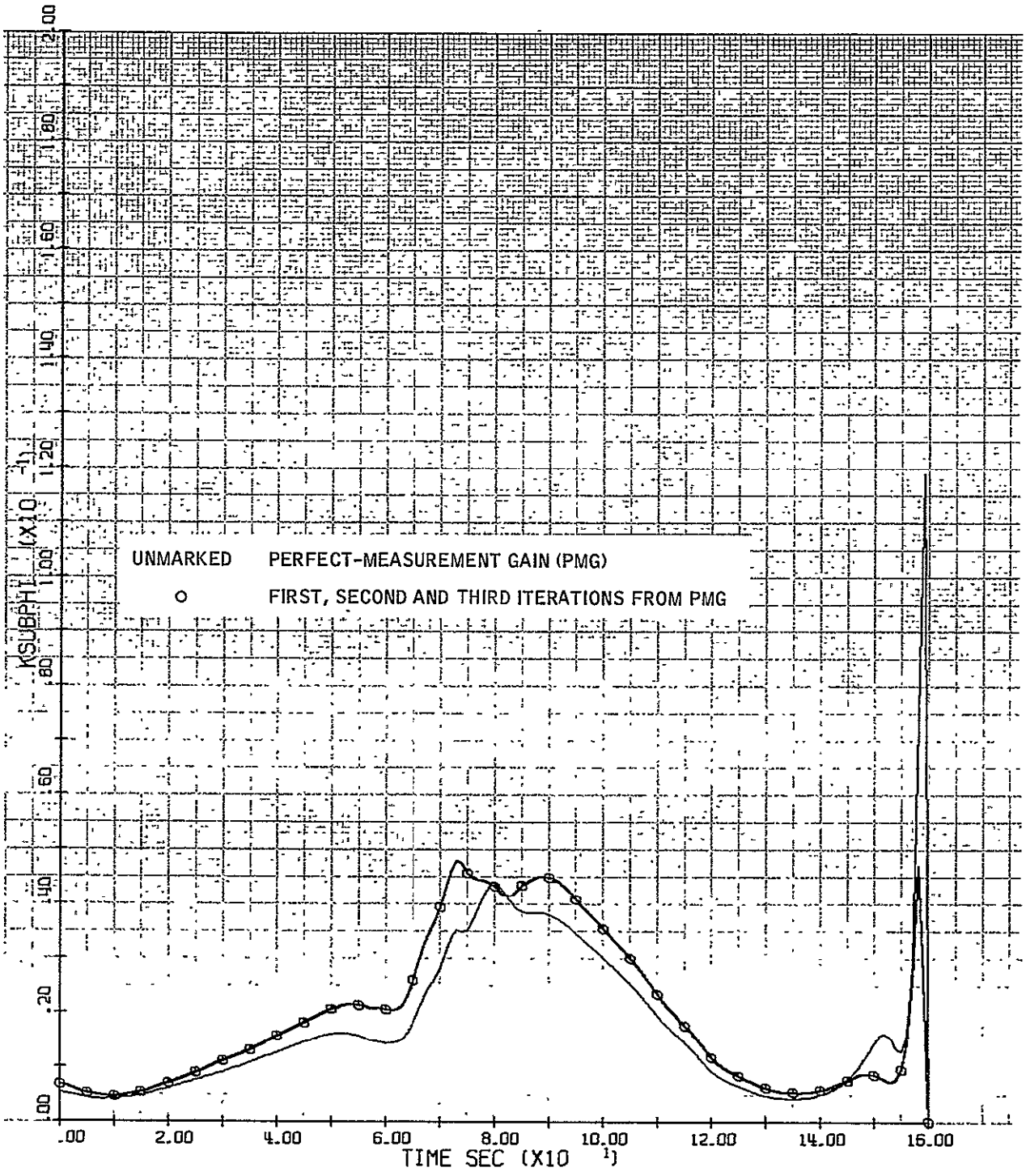


Figure C4. ϕ Gain, Iterations from Perfect-Measurement Gain

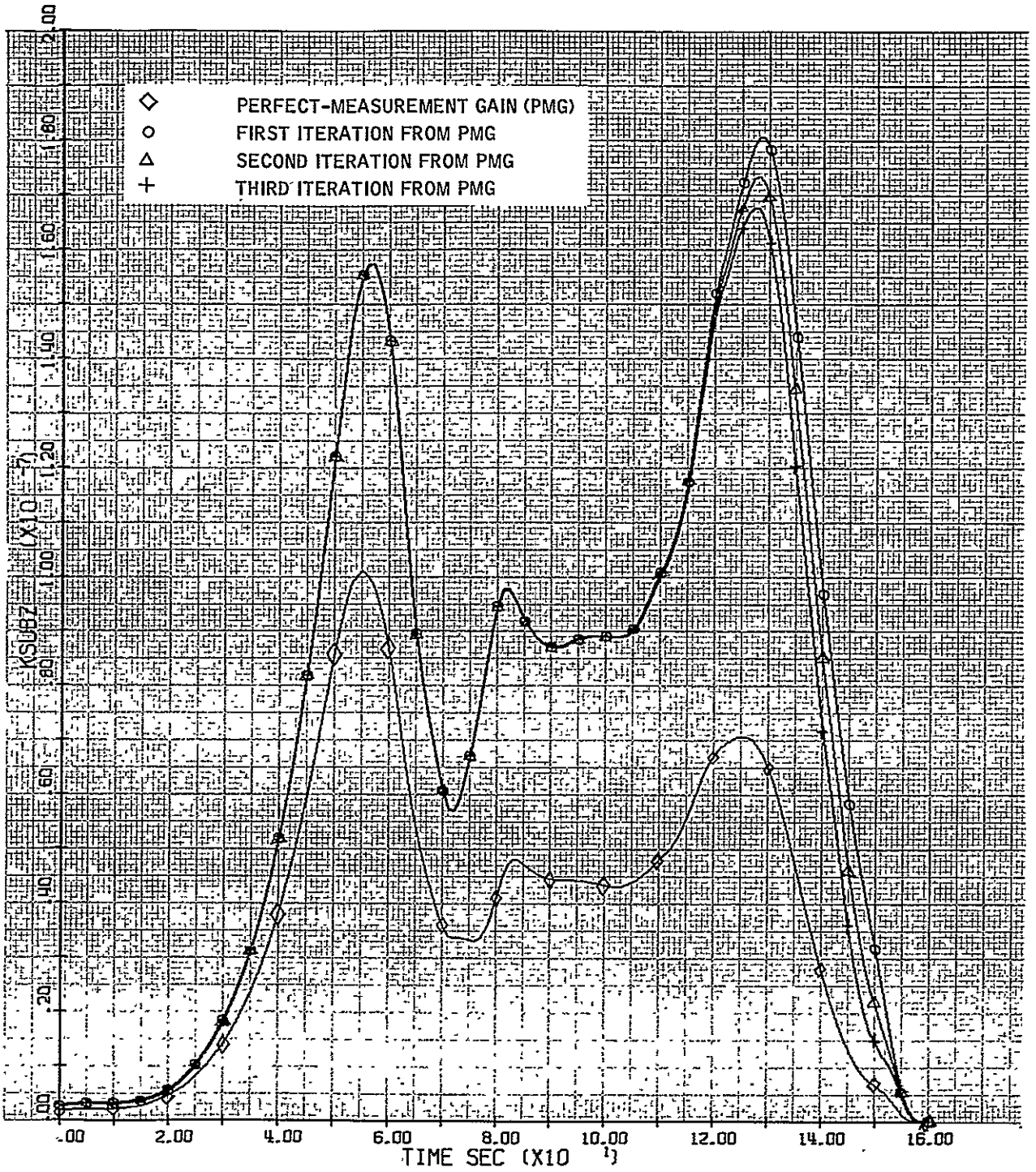


Figure C5. z Gain, Iterations from Perfect-Measurement Gain

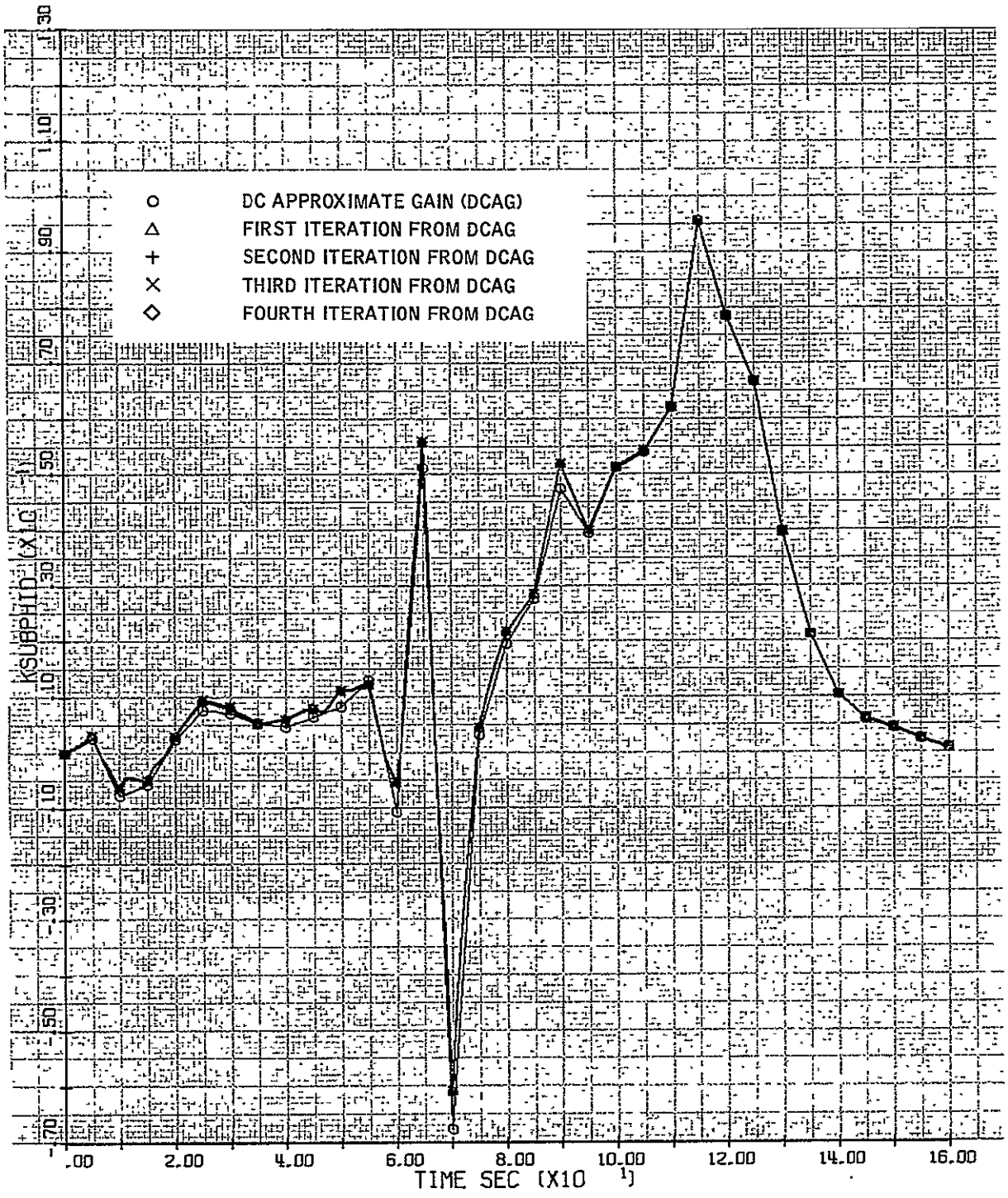


Figure C6. $\dot{\phi}$ Gain, Iterations from DC Approximation

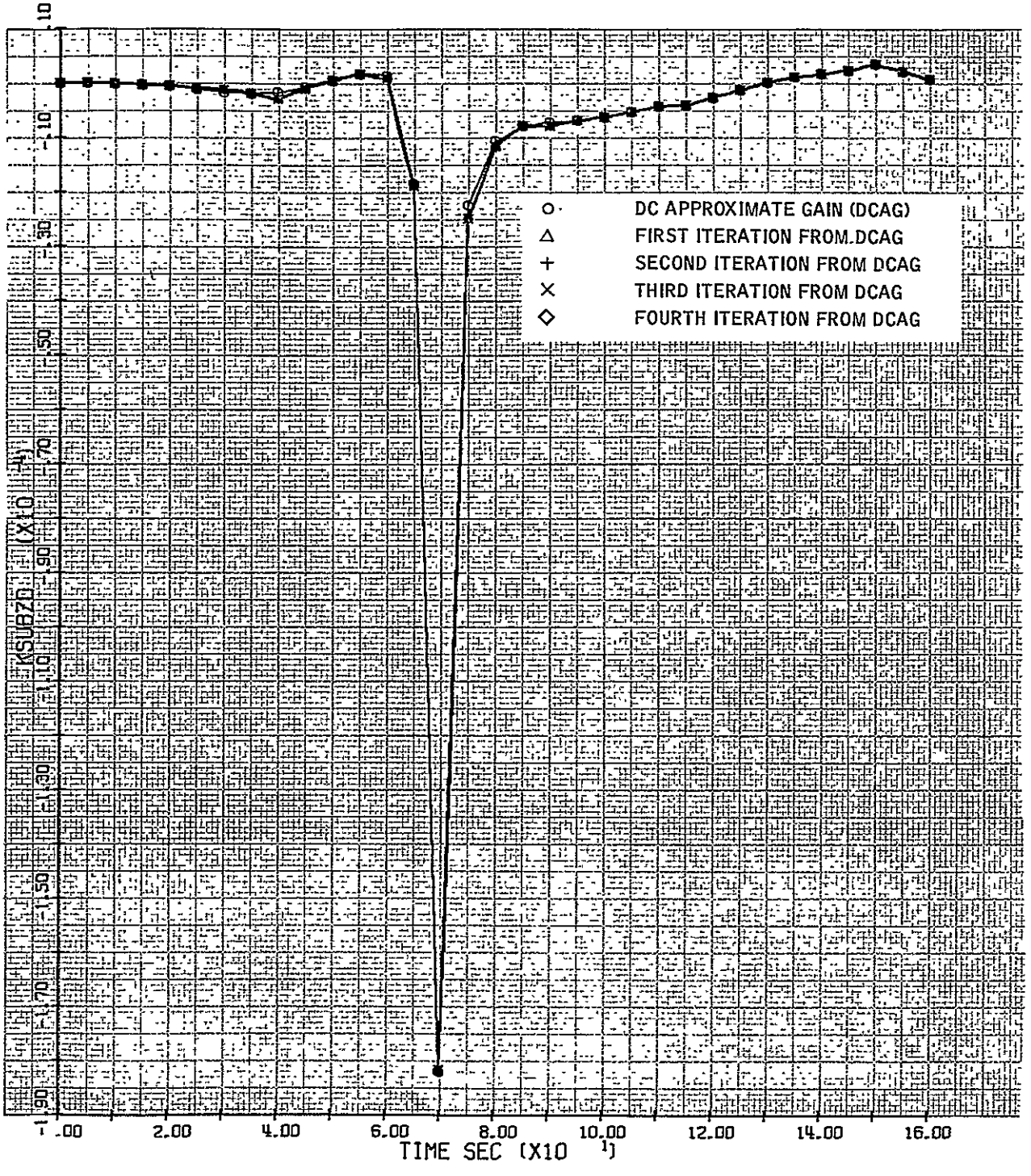


Figure C7. $\dot{z}$ Gain, Iterations from DC Approximation

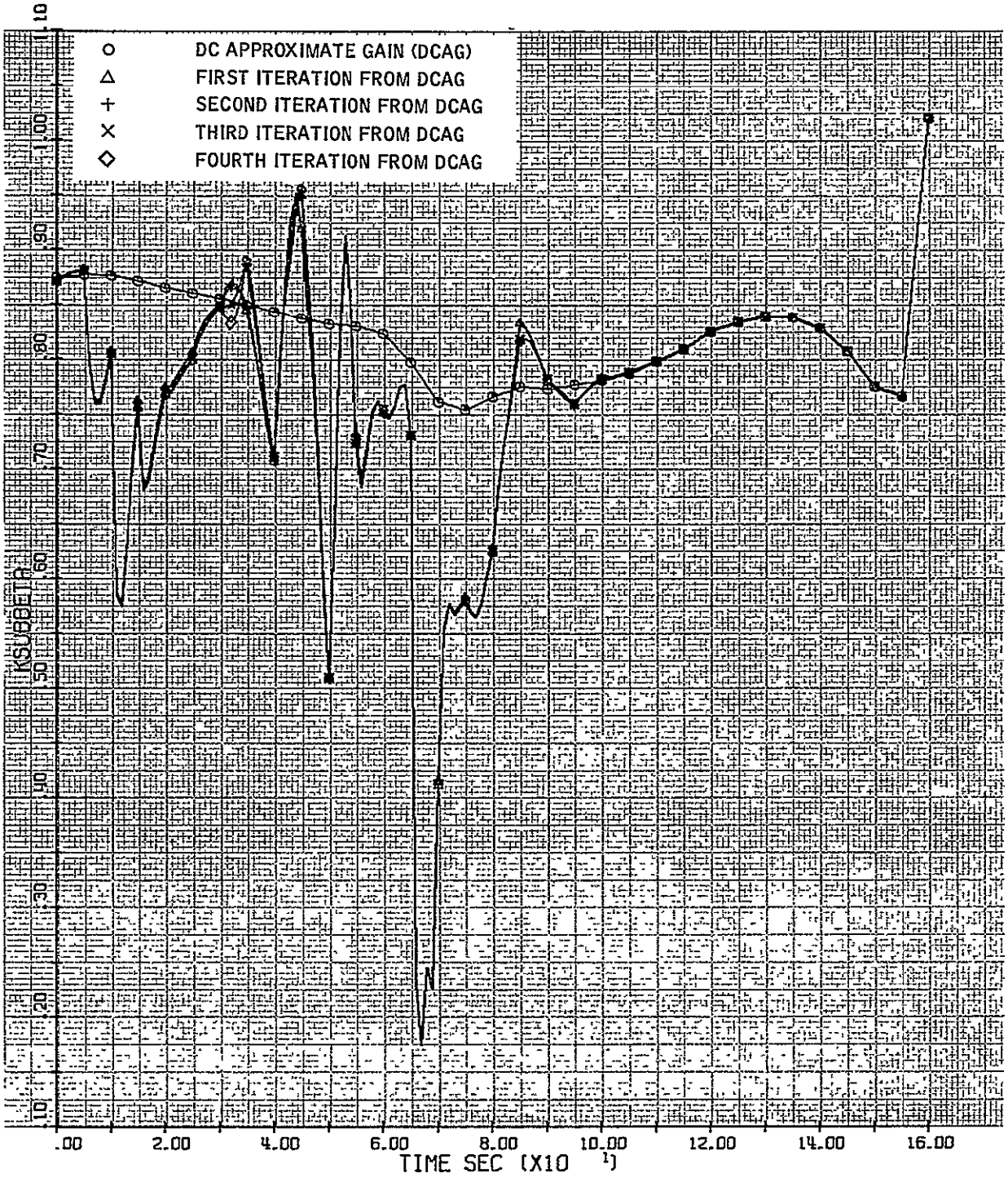


Figure C8. β Gain, Iterations from DC Approximation

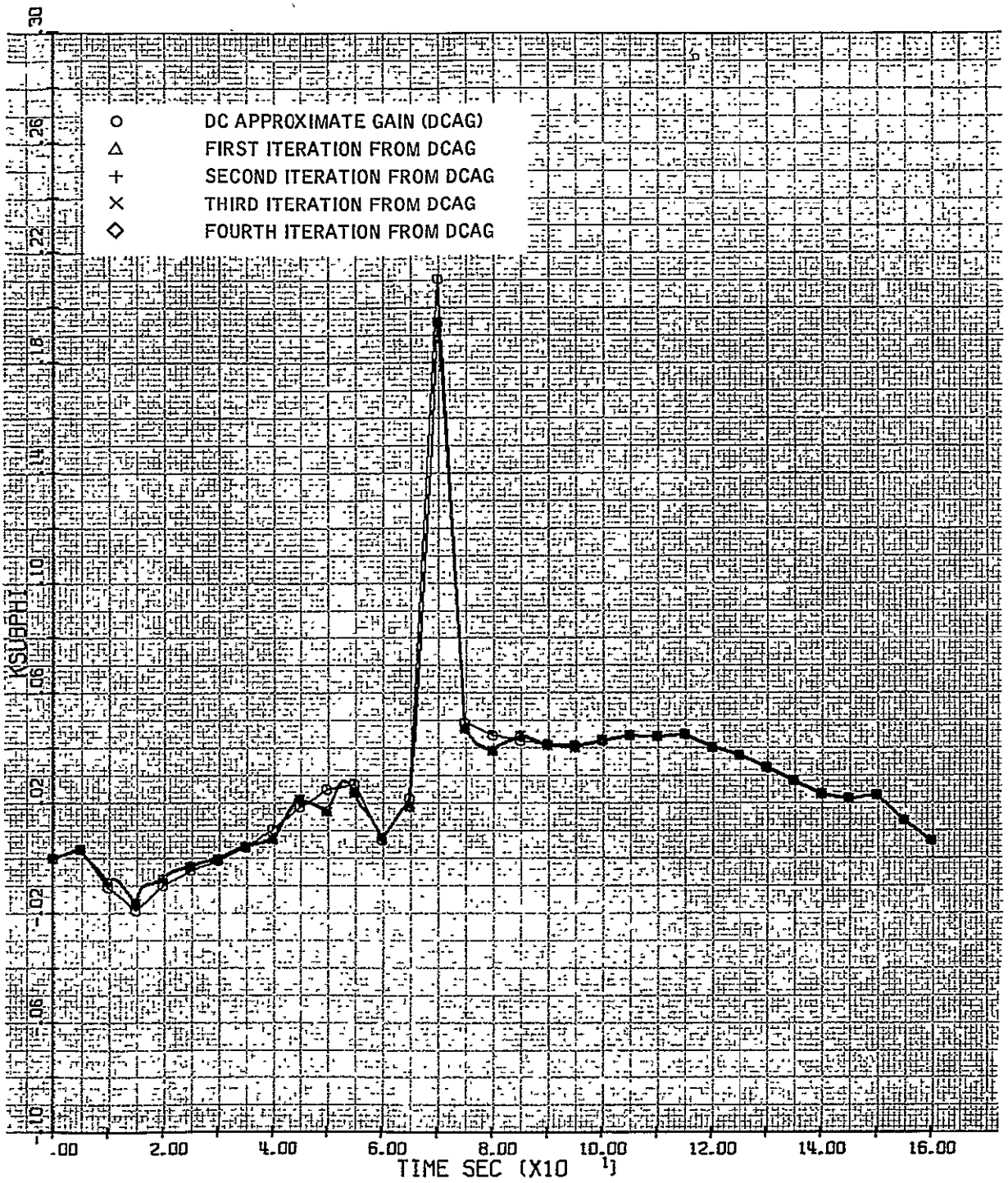


Figure C9. ϕ Gain, Iterations from DC Approximation

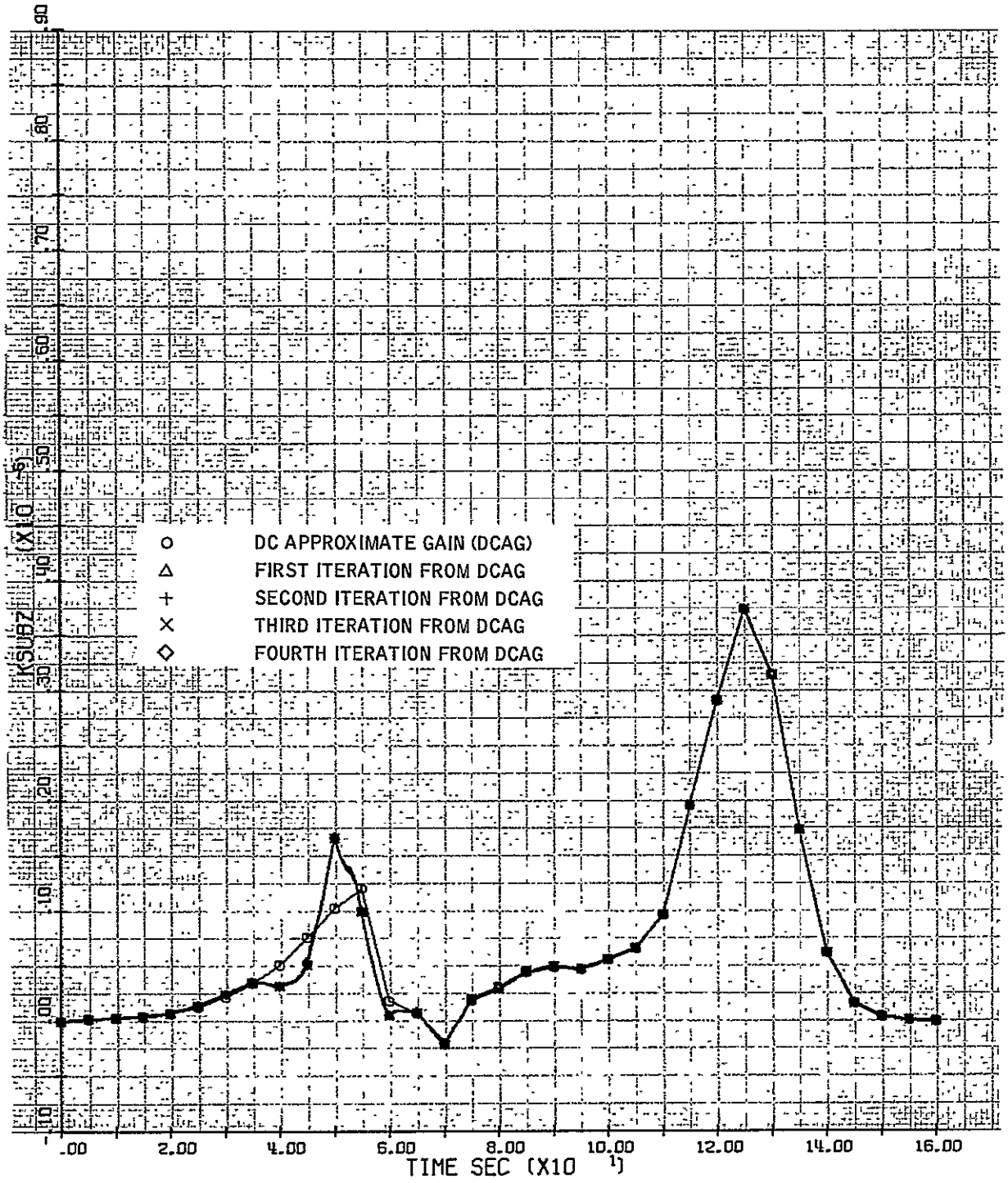


Figure C10. z Gain, Iterations from DC Approximation

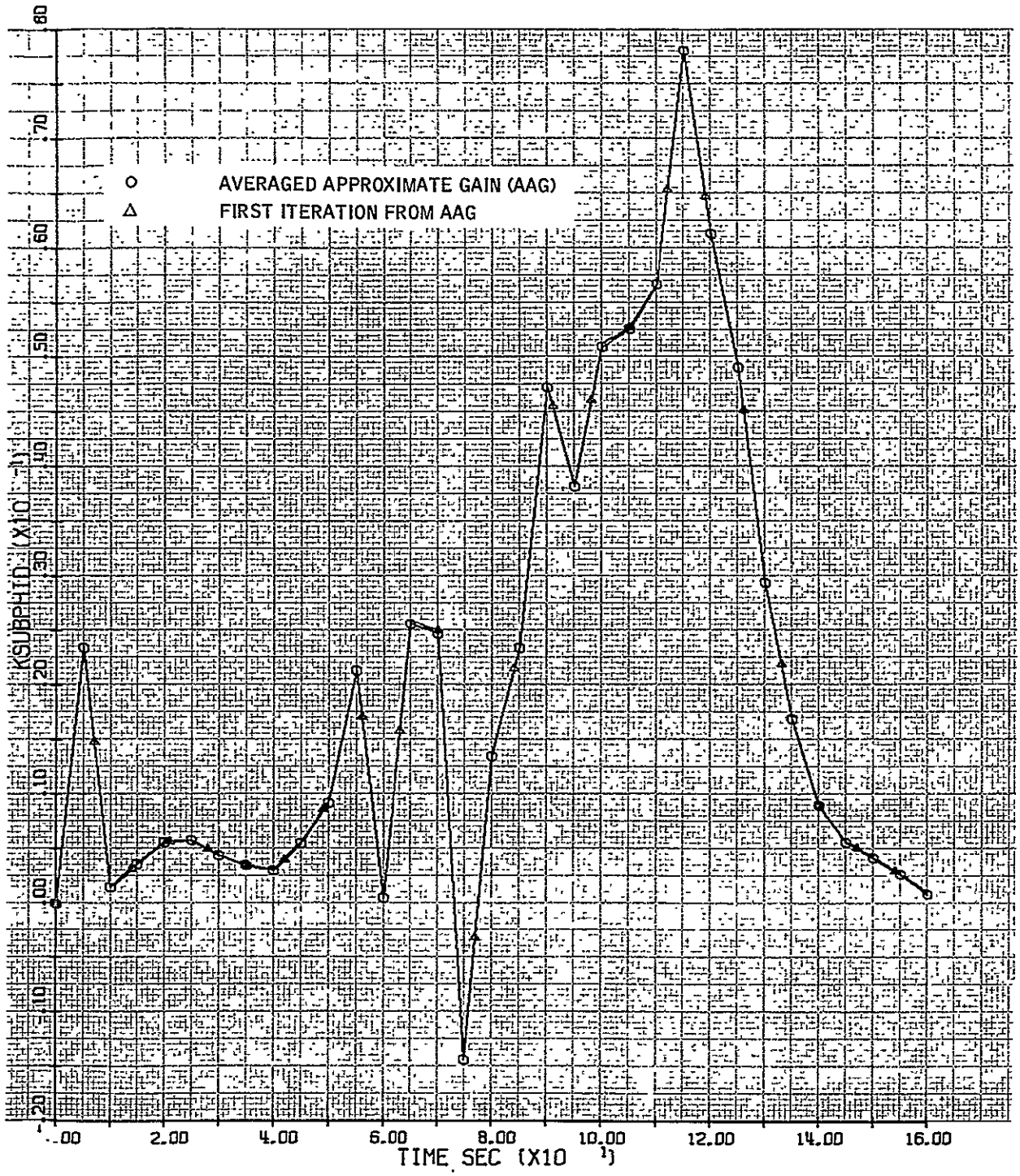


Figure C11. $\dot{\phi}$ Gain, Iterations from Averaged Approximation

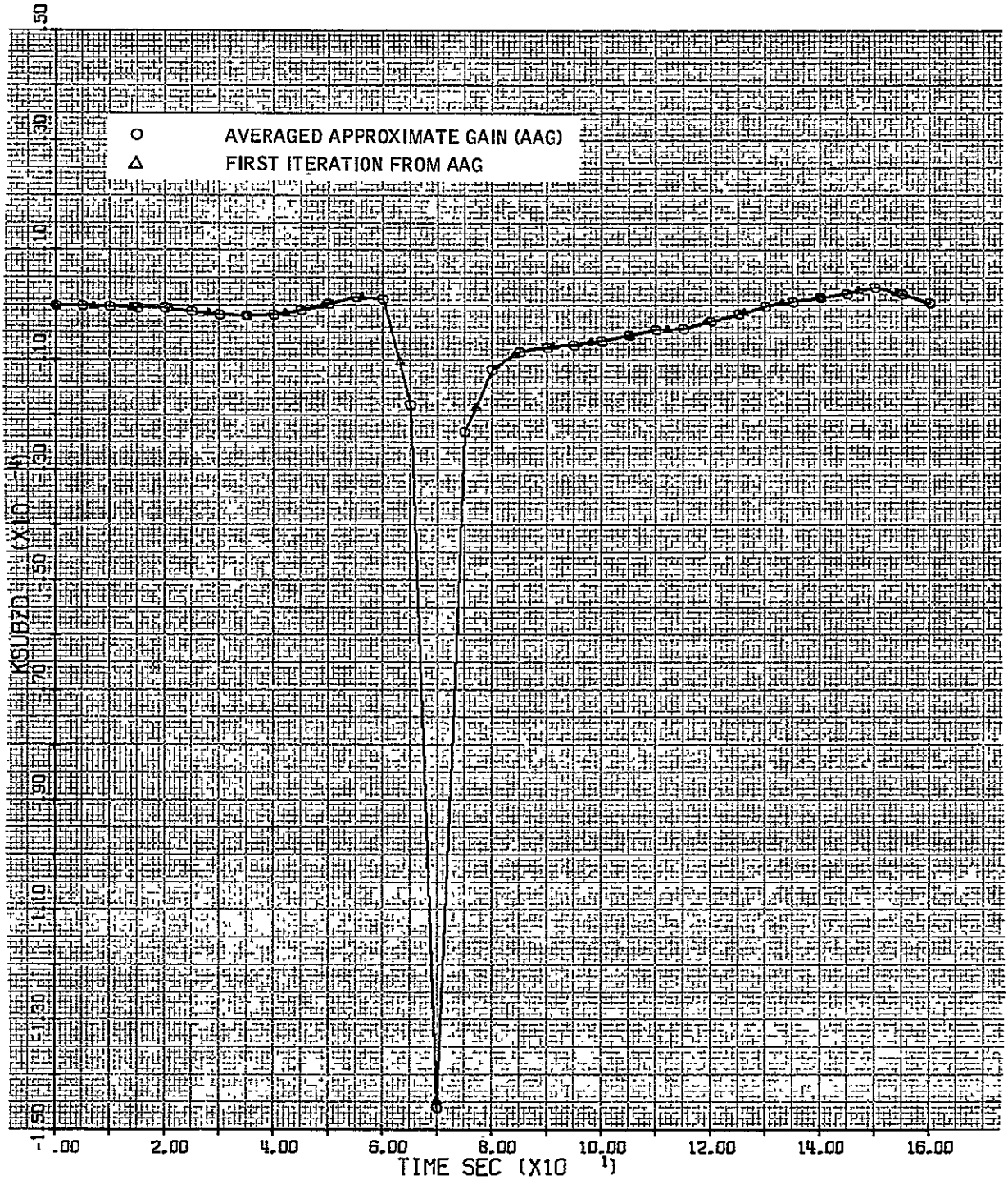


Figure C12. $\dot{z}$ Gain, Iterations from Averaged Approximation

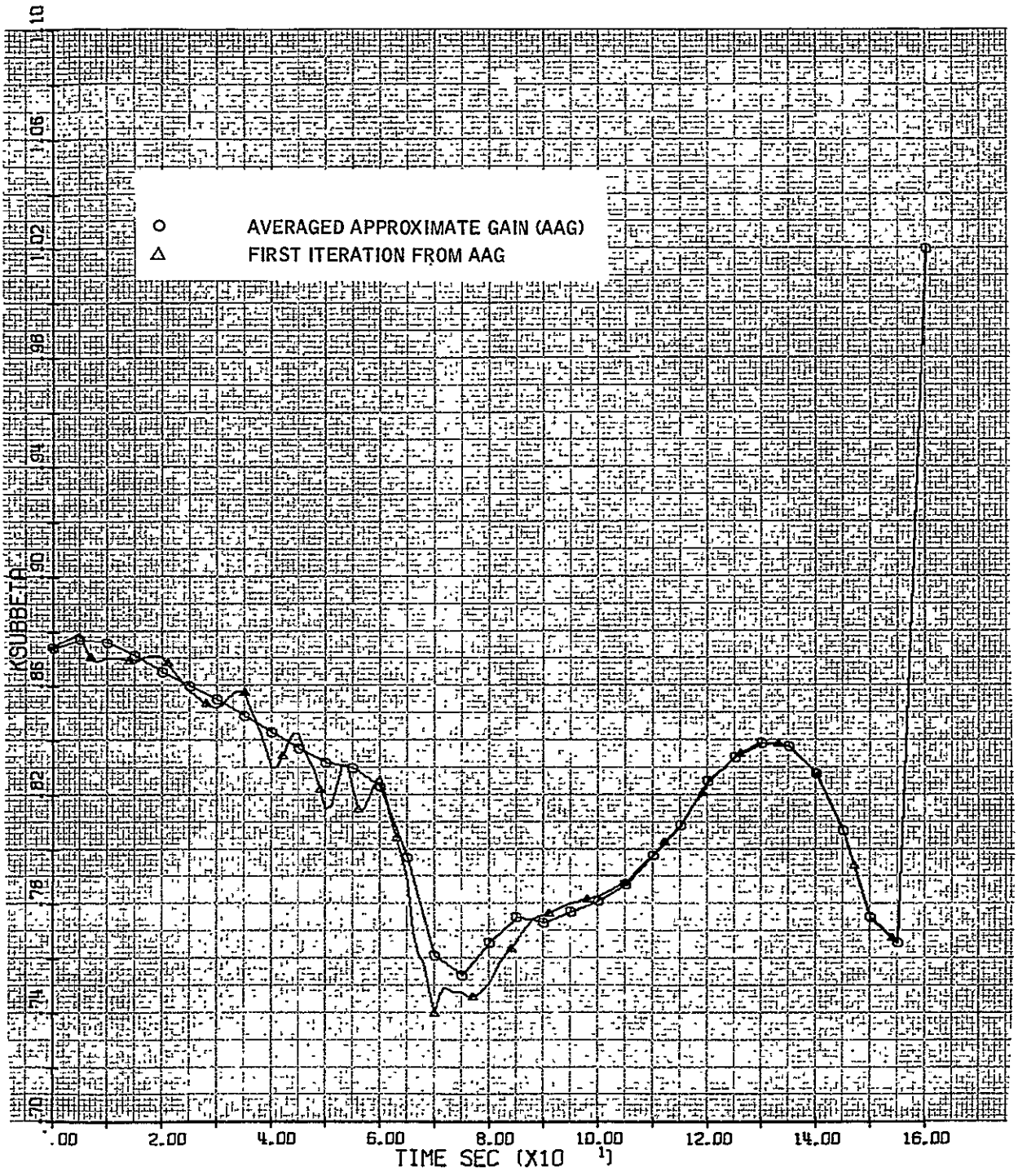


Figure C13. β Gain, Iterations from Averaged Approximation

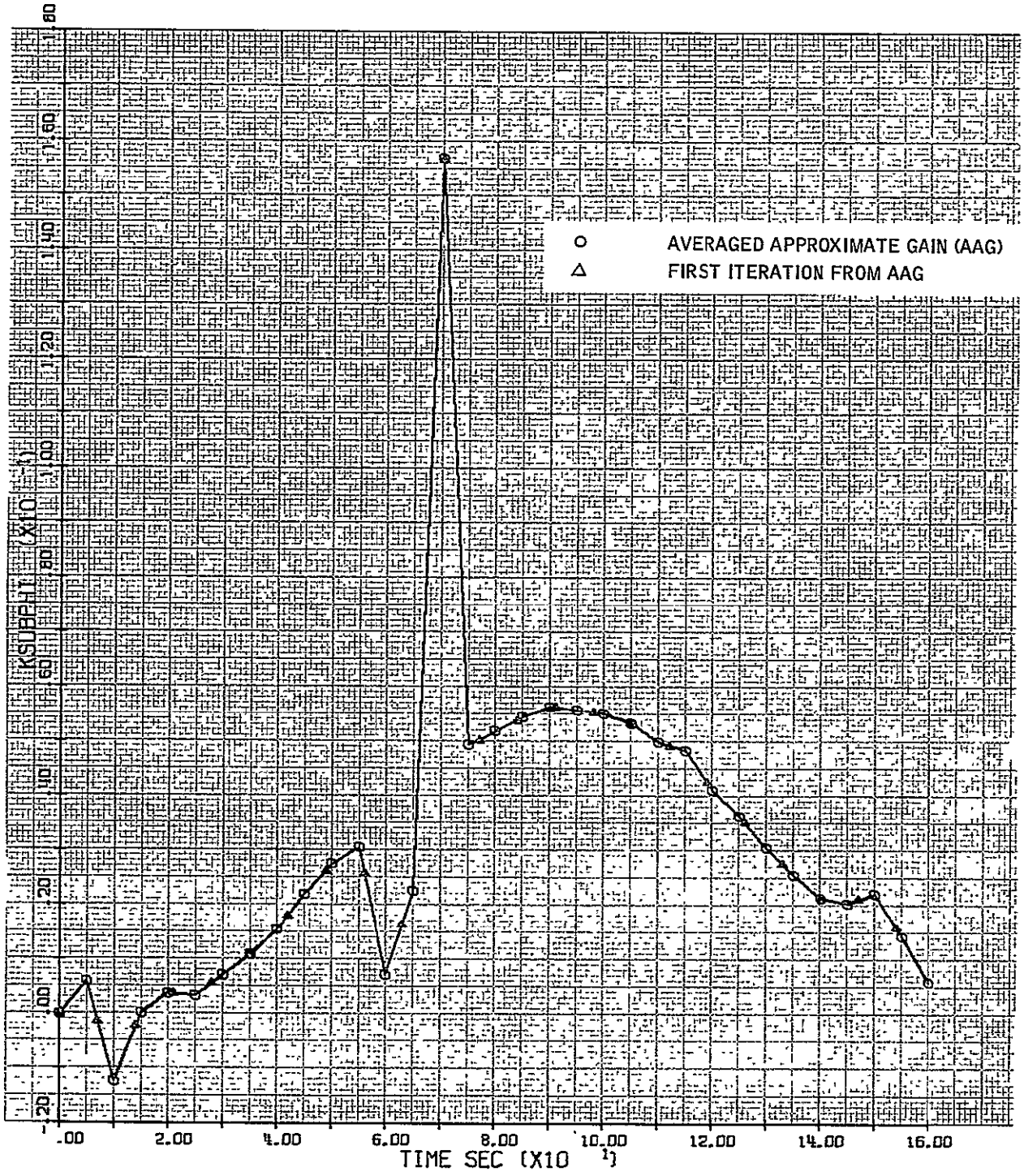


Figure C14. ϕ Gain, Iterations from Averaged Approximation

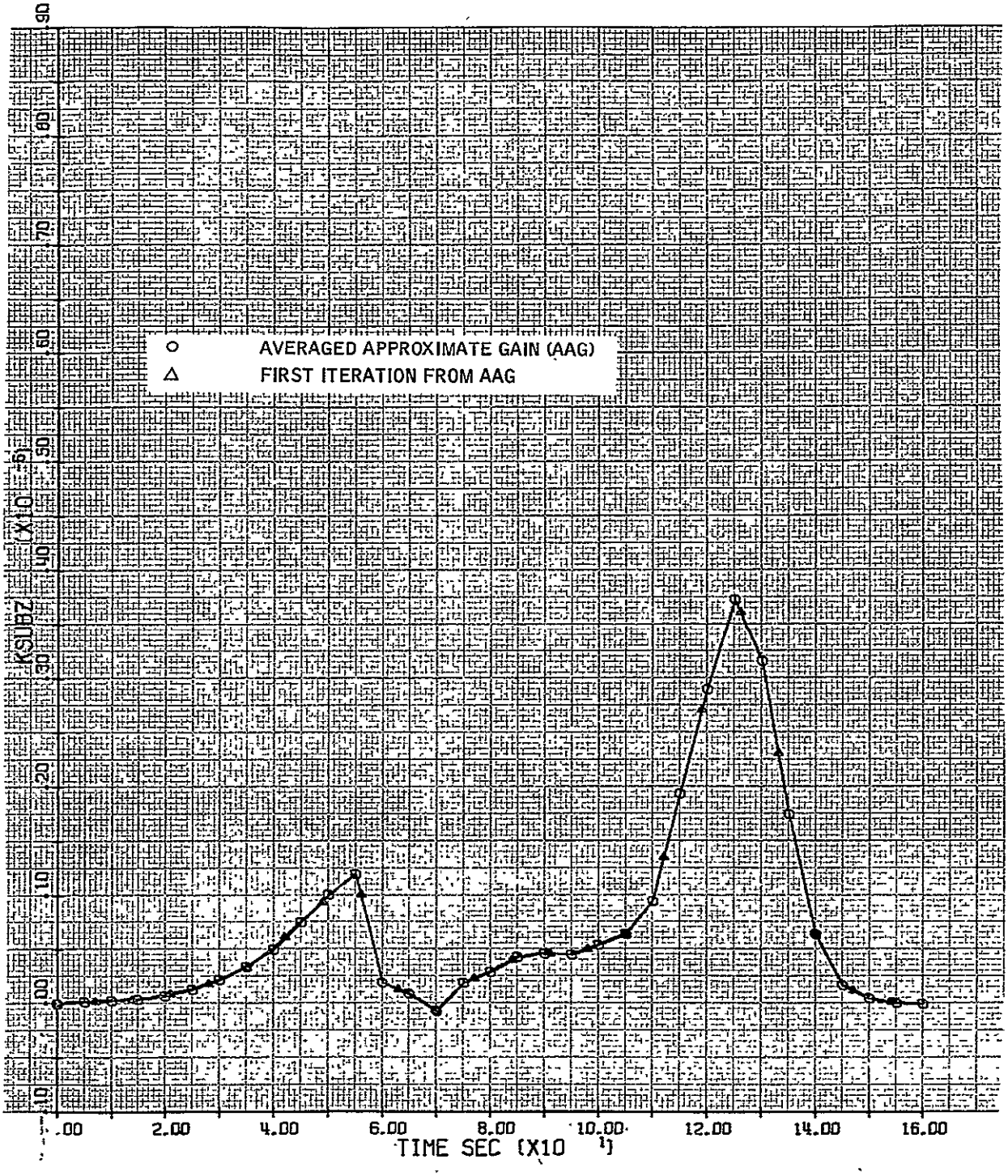


Figure C15. z Gain, Iterations from Averaged Approximation

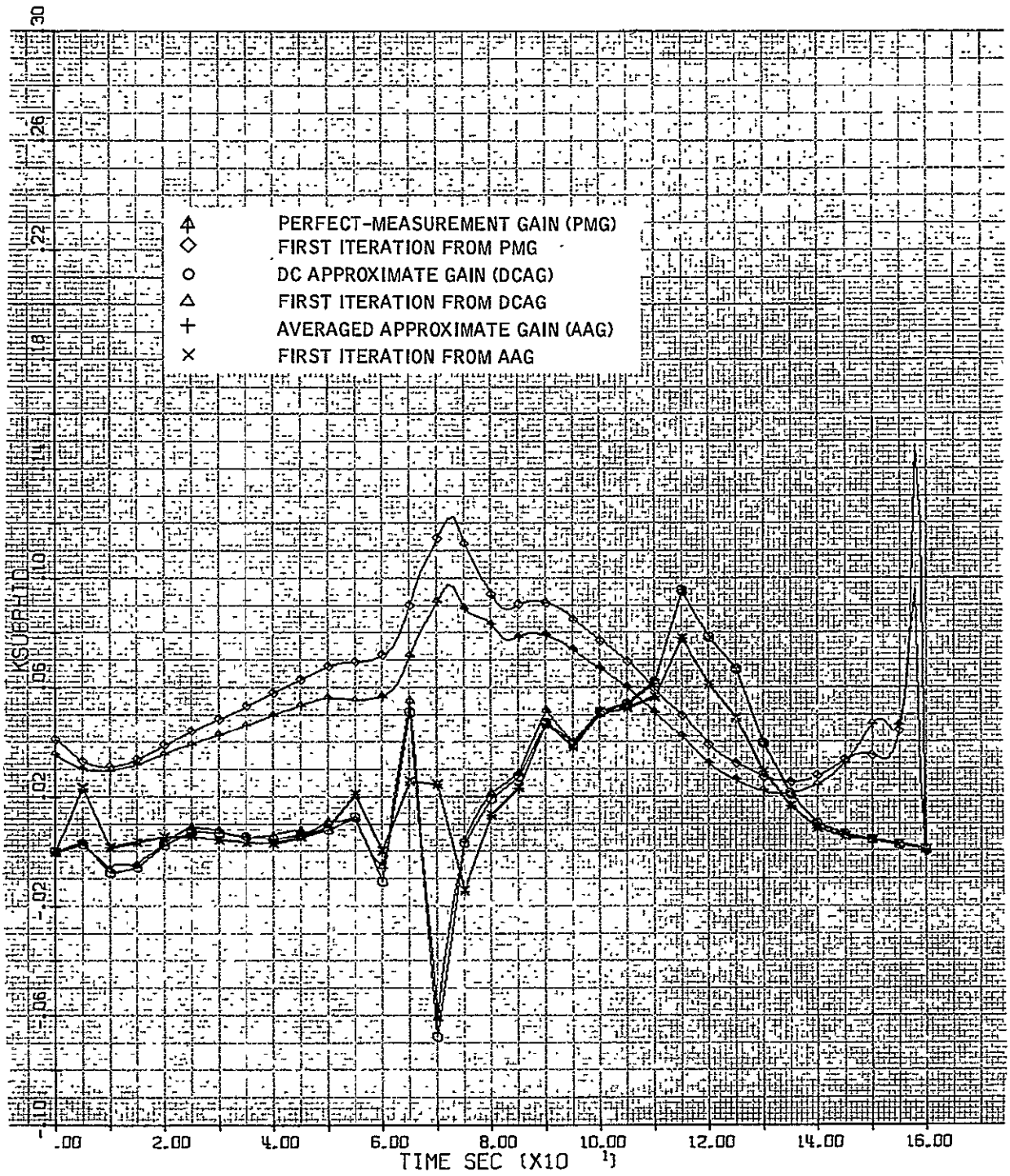


Figure C16. $\dot{\phi}$ Gain, Initial and First Iterations

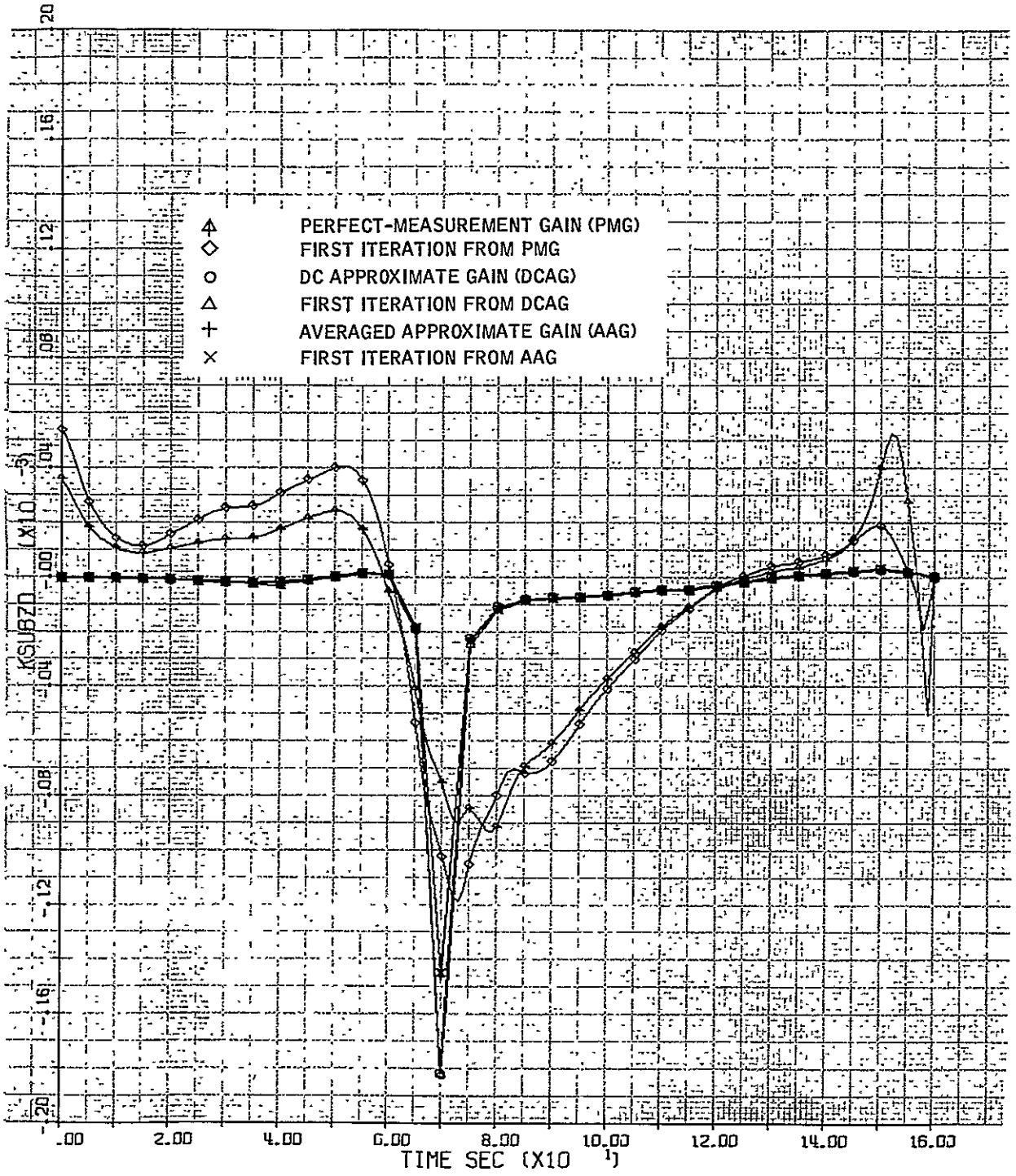


Figure C17. $\dot{z}$ Gain, Initial and First Iterations

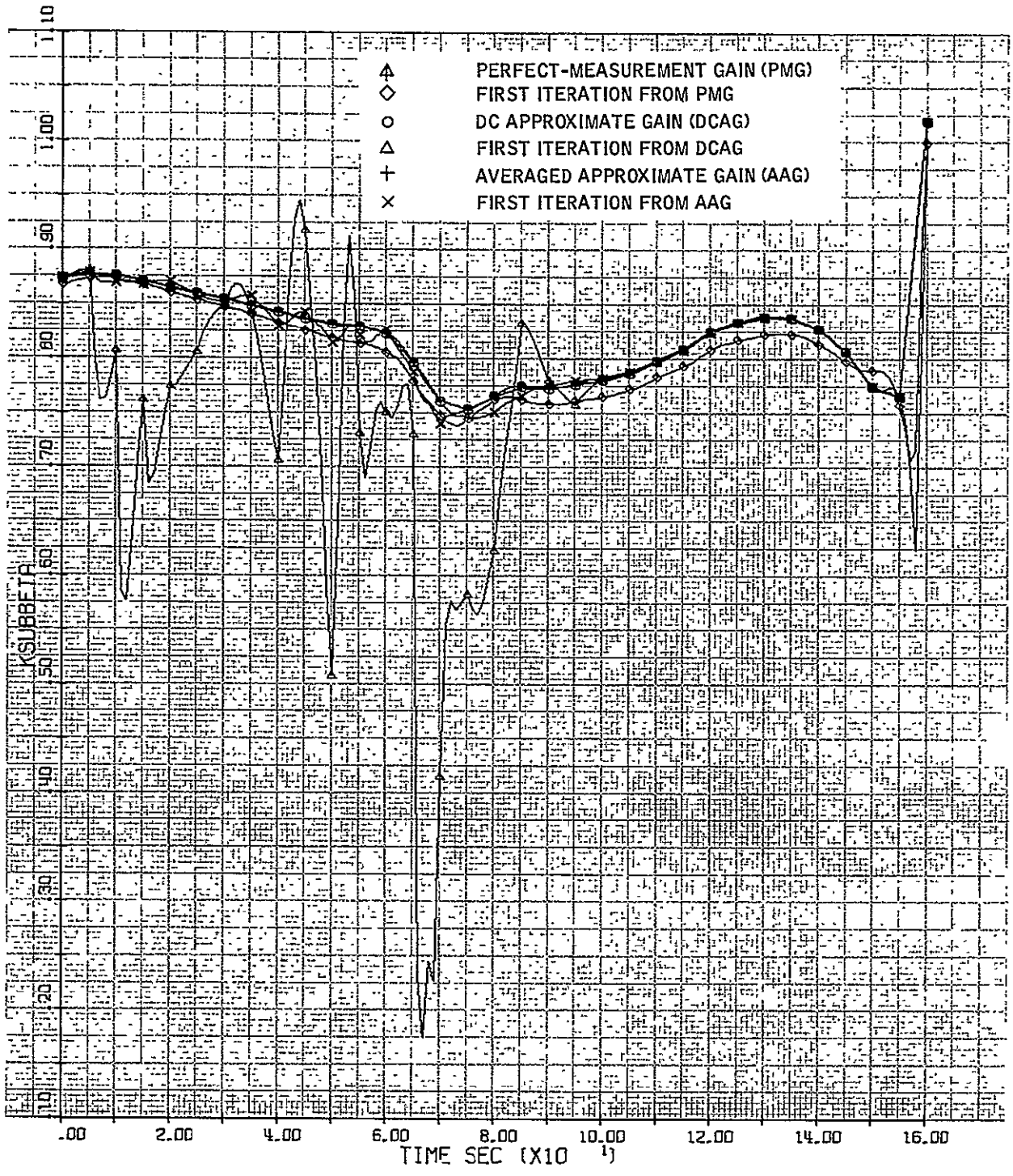


Figure C18. β Gain, Initial and First Iterations

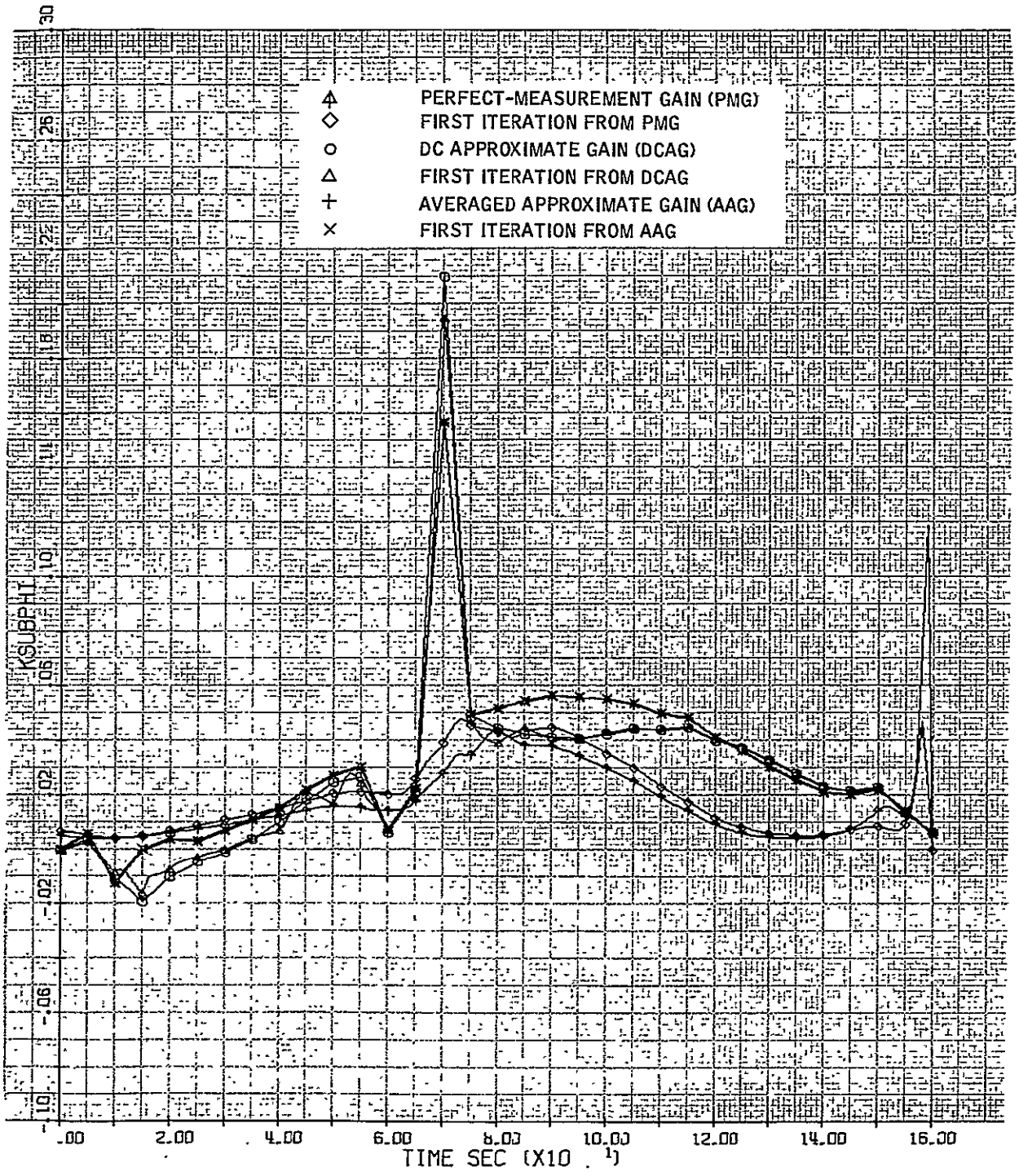


Figure C19. ϕ Gain, Initial and First Iterations

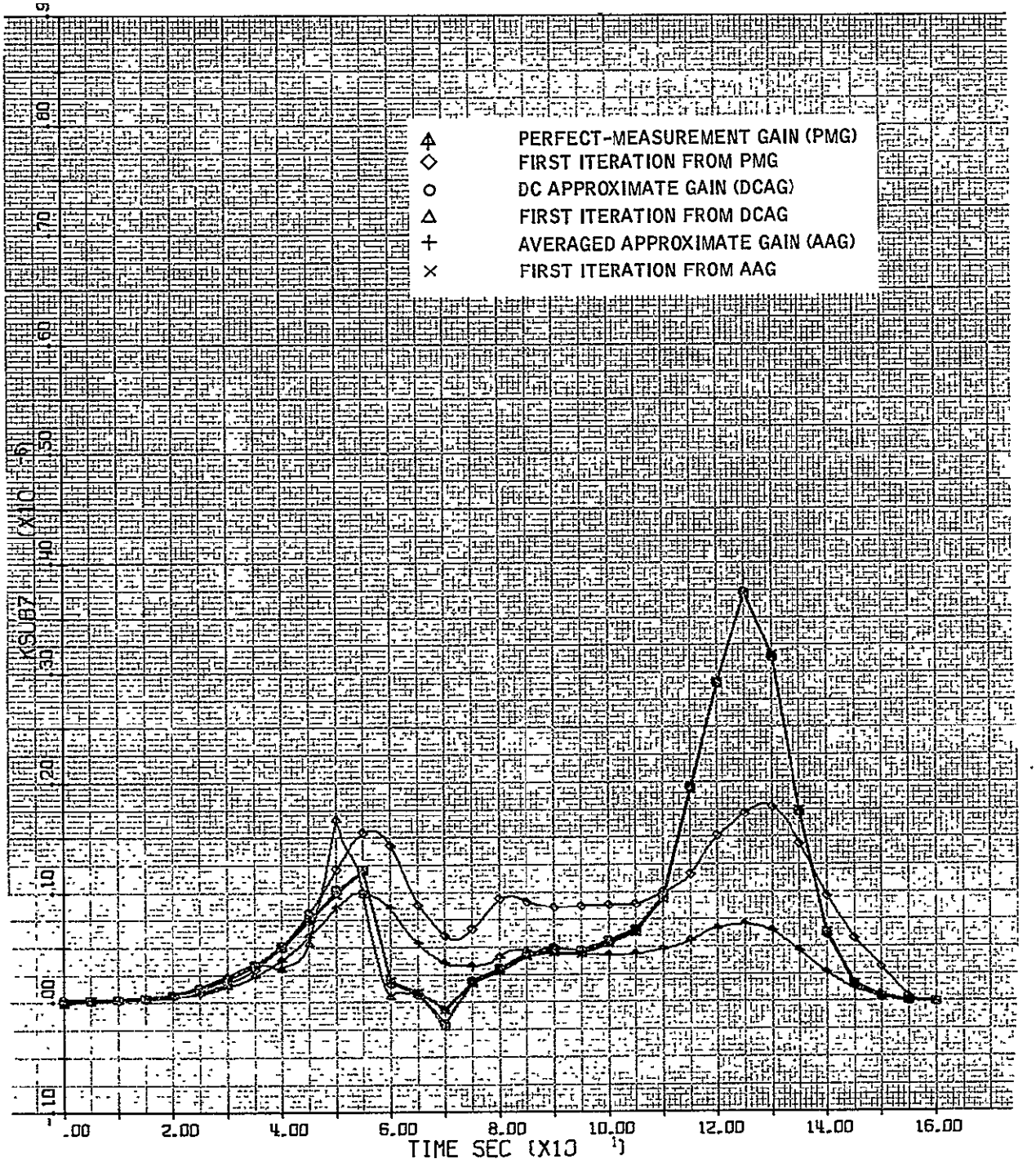


Figure C20. z Gain, Initial and First Iterations

APPENDIX D
COMPUTATION OF SENSITIVITY COEFFICIENTS

In this appendix three computational methods of determining sensitivity coefficients of the performance index for parameter variations of the system are presented.

Let the system equation be

$$\dot{x} = F(p)\, x + G(p)\, u \; ; \; x(o) = x_o \tag{D1}$$

where F is a nxn matrix,

G is a nxr matrix,
p is an m-vector of parameters

Let the performance index be

$$J = \int_o^\infty (x^T Q x + u^T R u) dt \tag{D2}$$

where Q is a nxn positive definite matrix, and R is a rxr positive definite matrix. Let the control law be

$$u = Kx \tag{D3}$$

where K is a rxn feedback gain matrix. Then the optimal feedback gain matrix K* which minimizes J, becomes

$$K^* = -R^{-1} G^T(p) P \tag{D4}$$

where P satisfies the following Riccati matrix algebraic equation

$$-PF - F^T P + PGR^{-1}G^T P - Q = 0 \tag{D5}$$

and the optimal performance index J* may be written as

$$J^* = \left\{ x_o^T P x_o \right\} = Tr \left\{ P X_o \right\} \tag{D6}$$

with X_o denoting $x_o x_o^T$.

On the other hand let us assume that K is simply a feedback gain matrix which stabilizes the system (D1). Then the corresponding system equation becomes

$$\dot{x} = (F + GK)\, x \; ; \qquad x(o) = x_o \tag{D7}$$

and the performance index is

$$J = \int_o^\infty x_o^T [e^{(F+GK)^T t} \; (Q+K^T R K) e^{(F+GK)t}] x_o \, dt \tag{D8}$$

$$= x_o^T [\int_o^\infty e^{(F+GK)^T t} \; (Q+K^T R K) e^{(F+GK)t} dt] x_o$$

$$= x_o^T A x_o = Tr \left\{ A \, X_o \right\} \tag{D9}$$

where

$$A \triangleq \int_o^\infty e^{(F+GK)^T t} (Q+K^T R K) e^{(F+GK)t} dt \tag{D10}$$

and it satisfies

$$(F+GK)^T A + A(F+GK) + (Q+K^T R K) = 0 \tag{D11}$$

Note that P is a function of only the parameter p, whereas A is a function of both the parameter p and K.

Now we can define the performance index error due to using a simple stabilization matrix K rather than the optimal feedback gain K* for any value of the parameter p as follows.

$$\begin{aligned} e(K,p) &= J(K,p) - J^*(p) \\ &= x_o^T A x_o - x_o^T P x_o \\ &= \mathrm{Tr}\left\{(A - P)\, X_o\right\} \\ &= \mathrm{Tr}\left[\left\{A(K,p) - P(p)\right\} X_o\right] \end{aligned} \tag{D12}$$

Note that if $K = K^* = -R^{-1}G^T P$, then A equals P and e(K, p) is zero.

The function e(K, p) can be expanded into a Taylor's series in terms of K and p at any nominal values of K_o and p_o as:

$$\begin{aligned} e(K,p) = {} & e(K_o,p_o) + \nabla_K e[K-K_o] + \nabla_p e[p-p_o] \\ & + \tfrac{1}{2}[K-K_o]^T \nabla_{Kp} e[p-p_o] + \tfrac{1}{2}[p-p_o]^T \nabla_{pK} e[K-K_o] \\ & + \tfrac{1}{2}[K-K_o]^T \nabla_{KK} e[K-K_o] + \tfrac{1}{2}[p-p_o]^T \nabla_{pp} e[p-p_o] \\ & + \text{higher order terms} \end{aligned} \tag{D13}$$

where K and $[K-K_o]$ are considered as vectors and all partial derivatives are evaluated at (K_o, p_o), $\nabla_K e$ is the (rn) row vector

$$\nabla_K e = \left[\frac{\partial e}{\partial K_{11}}, \frac{\partial e}{\partial K_{21}}, \ldots \frac{\partial e}{\partial K_{r1}}, \frac{\partial e}{\partial K_{12}}, \ldots, \frac{\partial e}{\partial K_{r2}}, \ldots, \frac{\partial e}{\partial K_{rn}}\right]$$

and ∇_{Kp} is the (rn)x(m) matrix

$$\nabla_{Kp}e = \begin{bmatrix} \frac{\partial^2 e}{\partial K_{11}\partial p_1} & \frac{\partial^2 e}{\partial K_{11}\partial p_2} & \cdots & \frac{\partial^2 e}{\partial K_{11}\partial p_m} \\ \frac{\partial^2 e}{\partial K_{21}\partial p_1} & \frac{\partial^2 e}{\partial K_{21}\partial p_2} & \cdots & \frac{\partial^2 e}{\partial K_{21}\partial p_m} \\ \vdots & \vdots & & \vdots \\ \frac{\partial^2 e}{\partial K_{rn}\partial p_1} & \frac{\partial^2 e}{\partial K_{rn}\partial p_2} & \cdots & \frac{\partial^2 e}{\partial K_{rn}\partial p_m} \end{bmatrix}$$

Similarly other second-order partial matrices are defined.

Since $e = J(K, p) - J^*(p)$

$$= Tr\left[\left\{A(K, p) - P(p)\right\} X_o\right],$$

$$\frac{\partial e}{\partial k} = \frac{\partial J}{\partial k} - \frac{\partial J^*}{\partial k}$$

$$= Tr\left\{\left(\frac{\partial A}{\partial k} - \frac{\partial P}{\partial k}\right)X_o\right\}$$

$$= Tr\left\{\left(\frac{\partial A}{\partial k}\right) X_o\right\} \text{ where k denotes an element of K} \qquad \text{(D14)}$$

Thus, in order to obtain a Taylor's series expension of e, we must obtain the partials of A and P with respect to K and p. These partials are called sensitivity coefficients of performance indices.

Since A and P are implicit functions of K and p, there is no direct method to determine these partials. The partials can be determined by solving

Lyapunov type matrix equations generated by applying the implicit function theorem to (D5) and D11). Sample sensitivity coefficient matrix equations follow:

$$(F+GK)^T\left(\frac{\partial A}{\partial K_{ij}}\right)+\left(\frac{\partial A}{\partial K_{ij}}\right)(F+GK) = -\left\{[AG+K^TR](E_{ij})\right\}^T - \left\{[AG+K^TR](E_{ij})\right\} \tag{D15}$$

where E_{ij} denotes an (r)x(n) matrix with the ij^{th} element equal to one and all other elements zero.

$$(F+GK)^T\left(\frac{\partial A}{\partial p_i}\right)+\left(\frac{\partial A}{\partial p_i}\right)(F+GK) = -\left(\frac{\partial F}{\partial p_i}+\frac{\partial G}{\partial p_i}K\right)^T A - A\left(\frac{\partial F}{\partial p_i}+\frac{\partial G}{\partial p_i}K\right) \tag{D16}$$

$$(F+GK^*)^T\left(\frac{\partial P}{\partial p_i}\right)+\left(\frac{\partial P}{\partial p_i}\right)(F+GK^*) = -\left(\frac{\partial F}{\partial p_i}+\frac{\partial G}{\partial p_i}K^*\right)^T P - P\left(\frac{\partial F}{\partial p_i}+\frac{\partial G}{\partial p_i}K^*\right) \tag{D17}$$

Similarly, second-order sensitivity coefficient equations are of the form

$$(F+GK)^T\left(\frac{\partial^2 A}{\partial K_{ij}\partial K_{km}}\right)+\left(\frac{\partial^2 A}{\partial K_{ij}\partial K_{km}}\right)(F+GK)$$

$$= -\left[\left(\frac{\partial A}{\partial K_{km}}\right)G(E_{ij})\right]^T - \left[\left(\frac{\partial A}{\partial K_{km}}\right)G(E_{ij})\right]$$

$$-\left[\left(\frac{\partial A}{\partial K_{ij}}\right)G(E_{km})\right]^T - \left[\left(\frac{\partial A}{\partial K_{ij}}\right)G(E_{km})\right]$$

$$-\left[(E_{ij})^T R(E_{km})\right]^T - \left[(E_{ij})^T R(E_{km})\right] \qquad \text{(D18)}$$

$$(F+GK)^T\left(\frac{\partial^2 A}{\partial K_{ij}\partial p_k}\right)+\left(\frac{\partial^2 A}{\partial K_{ij}\partial p_k}\right)(F+GK)$$

$$= -(GE_{ij})^T\frac{\partial A}{\partial p_k} - \frac{\partial A}{\partial p_k}GE_{ij}$$

$$-\left(\frac{\partial F}{\partial p_k}+\frac{\partial G}{\partial p_k}K\right)^T\frac{\partial A}{\partial K_{ij}} - \frac{\partial A}{\partial K_{ij}}\left(\frac{\partial F}{\partial p_k}+\frac{\partial G}{\partial p_k}K\right)$$

$$-\left(\frac{\partial G}{\partial p_k}E_{ij}\right)^T A - A\frac{\partial G}{\partial p_k}E_{ij} \qquad \text{(D19)}$$

Three methods for computing these partials are now discussed.

ADJOINT METHOD

Let the adjoint matrix equation be

$$(F+GK)W + W(F+GK)^T = X_o \qquad \text{(D20)}$$

Since (F+GK) is a stable system matrix and X_o is a positive semidefinite matrix, W is a negative semidefinite matrix.

Consider

$$\frac{\partial e}{\partial K_{ij}} = \mathrm{Tr}\left\{\frac{\partial A}{\partial K_{ij}} X_o\right\} \tag{D21}$$

Substituting (D20) into (D21) produces

$$\frac{\partial e}{\partial K_{ij}} = \mathrm{Tr}\left(\frac{\partial A}{\partial K_{ij}}\right)\left[(F+GK)W + W(F+GK)^T\right] \tag{D22}$$

Using the trace properties, $\mathrm{Tr}(AB) = \mathrm{Tr}(BA)$, $\mathrm{Tr}(A^T) = \mathrm{Tr}(A)$, we obtain

$$\frac{\partial e}{\partial K_{ij}} = \mathrm{Tr}\left\{\left[(F+GK)W + W(F+GK)^T\right]\left(\frac{\partial A}{\partial K_{ij}}\right)\right\}$$

$$= \mathrm{Tr}\left\{\left[(F+GK)W\left(\frac{\partial A}{\partial K_{ij}}\right) + W(F+GK)^T\left(\frac{\partial A}{\partial K_{ij}}\right)\right]\right\}$$

$$= \mathrm{Tr}\left\{\left[\left(\frac{\partial A}{\partial K_{ij}}\right)(F+GK)W + (F+GK)^T\left(\frac{\partial A}{\partial K_{ij}}\right)W\right]\right\}$$

$$= \mathrm{Tr}\left\{\left[(F+GK)^T\left(\frac{\partial A}{\partial K_{ij}}\right) + \left(\frac{\partial A}{\partial K_{ij}}\right)(F+GK)\right]W\right\}$$

The expression inside the bracket of (D23) is the left side of (D15); hence

$$\frac{\partial e}{\partial K_{ij}} = -\mathrm{Tr}\left[\left\{\left[(AG+K^TR)(E_{ij})\right]^T + \left[(AG+K^TR)(E_{ij})\right]\right\}W\right] \tag{D24}$$

As we can see, the right side of (D24) is known and hence $\frac{\partial e}{\partial K_{ij}}$ can be determined for all i, j; i = 1, ..., r; j = 1, ..., n.

Define another adjoint matrix V by the equation

$$(F+GK^*)V + V(F+GK^*)^T = X_o \tag{D25}$$

Then

$$\mathrm{Tr}\left[\frac{\partial P}{\partial p_k} X_o\right] = \mathrm{Tr}\left[\frac{\partial P}{\partial p_k}(F+GK^*)V + \frac{\partial P}{\partial p_k} V(F+GK^*)^T\right]$$

$$= \mathrm{Tr}\left[\frac{\partial P}{\partial p_k}(F+GK^*)V + (F+GK^*)^T \frac{\partial P}{\partial p_k} V\right]$$

$$= \mathrm{Tr}\left\{\left[\frac{\partial P}{\partial p_k}(F+GK^*) + (F+GK^*)^T \frac{\partial P}{\partial p_k}\right] V\right\} \tag{D26}$$

Using (D17) in equation (D26) yields

$$\mathrm{Tr}\left[\frac{\partial P}{\partial p_k} X_o\right] = -\mathrm{Tr}\left\{\left[\left(\frac{\partial F}{\partial p_k} + \frac{\partial G}{\partial p_k} K^*\right)^T P + P\left(\frac{\partial F}{\partial p_k} + \frac{\partial G}{\partial p_k} K^*\right)\right] V\right\}$$

Therefore

$$\frac{\partial e}{\partial p_k} = \mathrm{Tr}\left[\left(\frac{\partial A}{\partial p_k} - \frac{\partial P}{\partial p_k}\right) X_o\right]$$

$$= -\mathrm{Tr}\left\{\left[\left(\frac{\partial F}{\partial p_k} + \frac{\partial G}{\partial p_k} K\right)^T A + A\left(\frac{\partial F}{\partial p_k} + \frac{\partial G}{\partial p_k} K\right)\right] W\right\}$$

$$+\mathrm{Tr}\left\{\left[\left(\frac{\partial F}{\partial p_k} + \frac{\partial G}{\partial p_k} K^*\right)^T P + P\left(\frac{\partial F}{\partial p_k} + \frac{\partial G}{\partial p_k} K^*\right)\right] V\right\} \tag{D27}$$

The right-hand sides of (D24) and (D27) involve only known terms and V and W. Thus the first-order partials can be computed using (D20), (D24), (D25) and (D27). From equations (D24), (D26) and (D27) the following properties can be established:

Property 1: $$e(K, p_o)\Big|_{K=K^*(p_o)} = 0$$

Proof: $K=K^*$ implies $A = P$ since the equations (D5) and (D11) are identical for $K=K^*$. Therefore

$$e(K, p_o)\Big|_{K=K^*(p_o)} = \mathrm{Tr}[(A-P)X_o]\Big|_{A=P} = 0$$

Property 2: $$\nabla_K e\Big|_{K=K^*(p)} = 0$$

Proof: From (D24),

$$\frac{\partial e}{\partial K_{ij}}\Big|_{K=K^*} = -\mathrm{Tr}\left[\left(\left[(AG+K^TR)E_{ij}\right]^T + (AG+K^TR)E_{ij}\right)W\right]\Big|_{K=K^*=-R^{-1}G^TP}$$

$$= -\mathrm{Tr}\left[\left(\left[(AG-PG)E_{ij}\right]^T + (AG-PG)E_{ij}\right)W\right]\Big|_{A=P}$$

$$= 0$$

Property 3: $\nabla_p e \Big|_{K-K^*(p)} = 0$

Proof: From (D27)

$$\left.\frac{\partial e}{\partial p_k}\right|_{K=K^*(p)} = -\mathrm{Tr}\left[\left(\frac{\partial F}{\partial p_k} + \frac{\partial G}{\partial p_k}K^*\right)^T (AW-PV)\right]\Bigg|_{A=P}$$

$$-\mathrm{Tr}\left[(WA-VP)\left(\frac{\partial F}{\partial p_k} + \frac{\partial G}{\partial p_k}K^*\right)\right]\Bigg|_{A=P}$$

Which is zero if W=V. Comparison of (D20) and (D25) shows that K=K* implies W=V.

Equations for the second-order partial derivative matrices may be obtained using the adjoint matrices and equations of the form of (D18) and (D19). For example

$$\frac{\partial^2 e}{\partial K_{ij}\partial K_{km}} = \mathrm{Tr}\left(\frac{\partial^2 A}{\partial K_{ij}\partial K_{km}}X_o\right)$$

$$= -\mathrm{Tr}\left\{\left[\left[\left(\frac{\partial A}{\partial K_{km}}\right)G(E_{ij})\right]^T + \left[\left(\frac{\partial A}{\partial K_{km}}\right)G(E_{ij})\right]\right]W\right\}$$

$$-\mathrm{TR}\left\{\left[\left[\left(\frac{\partial A}{\partial K_{ij}}\right)G(E_{km})\right]^T + \left[\left(\frac{\partial A}{\partial K_{ij}}\right)G(E_{km})\right]\right]W\right\}$$

$$-\mathrm{Tr}\left\{[[(E_{ij})^T R(E_{km})]^T + [(E_{ij})^T R(E_{km})]]W\right\} \qquad \text{(D28)}$$

As we can see from (D28) we need $\left(\frac{\partial A}{\partial K_{ij}}\right)$ to determine $\frac{\partial^2 e}{\partial K_{ij}\partial K_{km}}$. The only way we can obtain $\left(\frac{\partial^2 e}{\partial K_{ij}}\right)$ is to solve the matrix equation (D15). We must solve rn matrix equations to have $\nabla^2_{KK} e$. This is a severe drawback for the adjoint method of determining the partials.

From (D28) we can obtain a direct proof of the following property of the second partial.

Property 4: $\nabla_{KK} e \Big|_{K=K^*}$ is a positive semidefinite matrix. It is positive definite if X_o is positive definite.

Proof: If K=K*, then the right-hand side of equation (D15) is zero.

Thus $\frac{\partial A}{\partial k}\Big|_{K=K^*} = 0$ where k denotes any element of K. Hence, from (D28)

$$\frac{\partial^2 e}{\partial K_{ij}\partial K_{km}}\Big|_{K=K^*} = -2\ \mathrm{Tr}[(E_{ij})^T R\ E_{km} W]$$

Direct calculation reveals that $\mathrm{Tr}[E_{ij}{}^T R\ E_{km} W] = r_{ik} w_{mj}$. The matrix W is symmetric so that $r_{ik} w_{mj} = r_{ik} w_{jm}$. Using the Kronecker product notation we can then write:

$$\nabla_{KK} e \Big|_{K=K^*} = -2\ R \times W$$

R is assumed to be positive definite and W is negative semidefinite (definite if X_o is definite). The eigenvalues of the matrix $R \times W$ are $\lambda_i \mu_j$ where λ_i are the eigenvalues of R and μ_j are the eigenvalues of W (Ref. D1). Since R and W are symmetric, $\lambda_i > 0$ and $\mu_j \leq 0$ for each i and j. Thus $\lambda_i \mu_j \leq 0$ so the eigenvalues of $-R \times W$ are greater than or equal to zero. Strict inequalities follow from the assumption that X_o is positive definite.

Next we will consider the second method of computing the partials which is a vector equation method.

VECTOR EQUATION METHOD

The major drawback of the first method is that solutions of many Lyapunov-type equations for different inputs (right sides) are required. Since the left side of the equation does not change, we may not need to solve the Lyapunov equation for each different right side. There have been numerous investigations which converted the Lyapunov equation into linear algebraic equations (Ref. D2, D3). We will use Bingulac's technique (Ref. D3).

Let the Lyapunov-type equation be

$$PA + A^T P = -Q$$

Then Bingulac showed how to convert this equation into the following linear algebraic equation:

$$BP_\nu = -Q_\nu$$

where

$$P_\nu^T = (P_{11}, P_{12}, \ldots, P_{1n}, P_{22}, \ldots, P_{2n}, P_{33}, \ldots, P_{n-1\ n-1}, P_{nn})$$

$$R_\nu^T = (q_{11}, q_{12}, \ldots, q_{1n}, q_{22}, \ldots, q_{2n}, q_{33}, \ldots, q_{n-1\ n-1}, q_{nn})$$

$B = \{b_{ij}\}$ is a mxn matrix depending on the elements of A and

$$m = \frac{n(n+1)}{2}.$$

Let us apply this technique to (D15).

$$(F+GK)^T \left(\frac{\partial A}{\partial K_{ij}}\right) + \left(\frac{\partial A}{\partial K_{ij}}\right) (F+GK) = -C$$

where $C = \left\{[AG+K^TR](E_{ij})\right\}^T + \left\{[AG+K^TR](E_{ij})\right\}$. The corresponding linear algebraic equation is

$$D \left(\frac{\partial A}{\partial K_{ij}}\right)_\nu = -C_\nu \tag{D29}$$

where D is a $\frac{n(n+1)}{2}$ x $\frac{n(n+1)}{2}$ matrix and its elements depend on the elements of (F+GK), and

$$\left(\frac{\partial A}{\partial K_{ij}}\right)_\nu^T = \left(\frac{\partial A_{11}}{\partial K_{ij}}, \frac{\partial A_{12}}{\partial K_{ij}}, \ldots, \frac{\partial A_{1n}}{\partial K_{ij}}, \frac{\partial A_{22}}{\partial K_{ij}}, \ldots, \frac{\partial A_{2n}}{\partial K_{ij}}, \frac{\partial A_{33}}{\partial K_{ij}}, \ldots, \frac{\partial A_{nn}}{\partial K_{ij}}\right)$$

$$C_\nu^T = (C_{11}, C_{12}, \ldots, C_{1n}, C_{22}, C_{23}, \ldots, C_{2n}, C_{33}, \ldots, C_{3n}, \ldots, C_{n-1\ n-1}, C_{nn}).$$

The equations corresponding to second partials of A may be expressed in the form of (D29) with the same D matrix. The right hand side (the C_ν's) would be linear functions of the first partials. Thus the second partials could be expressed in the form

$$D^{-1}\left[T_1 D^{-1} T_2 + T_3\right]$$

where the T_i are well defined matrices and vectors of appropriate dimensions. The elements of T_i depend on F, G, K, Q, R and the indices of the variables with respect to which derivatives are being taken. Thus, the second partials can be easily computed once D^{-1} is determined. However, this method has a different drawback in comparison with the first method, namely unreasonable computer memory requirements. For example, let the order of the system be 20. Then the number of elements of D is $[20(20+1)/2]^2 = 44,100$. This storage requirement is too severe.

The first method for computing the partials requires solutions of many Lyapunov equations, whereas the second method requires extensive computer memory. A third method, described below, is a compromised version of the two previous methods.

POWER SERIES METHOD

Jameson (Ref. D4) formulated a method of solving a Lyapunov-type equation, using a power series of A up to n^{th} power, without increasing the dimensions of the equations.

Let a Lyapunov-type equation be

$$AX + XA^T = C \qquad \text{(D30)}$$

Then X is expressed as

$$X = Z^{-1} \left[\sum_{i=0}^{n-1} \sum_{j=0}^{n-i-1} (-1)^{i+j} a_i \, A^{(n-i-1-j)} C \, (A^T)^{(j)} \right] \qquad \text{(D31)}$$

where

$$a_o = 1$$

$$Z = 2(-1)^n \left[\sum_{i=0}^{n/2} a_{n-2i} \, A^{2i} \right] \qquad ; A^o = I$$

$$a_n = -\frac{1}{n} \left[\sum_{i=0}^{n-1} a_i \, \mathrm{Tr}(A^{n-i}) \right]$$

Applying (D31) to $\left(\frac{\partial A}{\partial K_{km}} \right)$, we obtain

$$\left(\frac{\partial A}{\partial K_{km}} \right) = Z^{-1} \left[\sum_{i=0}^{n-1} \sum_{j=0}^{n-i-1} (-1)^{i+j} a_i \, (F+GK)^{(n-i-1-j)} \left\{ -1 (E_{km})^T \, [AG+K^T R]^T \right.\right.$$

$$\left.\left. - \, [AG+K^T R] \, (E_{km}) \right\} (F+GK)^{T^{(j)}} \right] \qquad \text{(D32)}$$

where

$$Z = 2(-1)^n \left[\sum_{i=0}^{n/2} a_{n-2i} \, (F+GK)^{2i} \right] ; \; (F+GK)^o = I$$

$$a_n = -\frac{1}{n} \left[\sum_{i=0}^{n-1} a_i \, \mathrm{Tr} \left\{ (F+GK)^{n-i} \right\} \right]$$

Note that Z^{-1} and a_n are fixed quantities for a given (F+GK). Then

$$\frac{\partial e}{\partial K_{km}} = Tr\left(\frac{\partial A}{\partial K_{km}} X_o\right)$$

$$= Tr[Z^{-1}\left[\sum_{i=0}^{n-1}\sum_{j=0}^{n-i-1}(-1)^{i+j}a_i\,(F+GK)^{(n-i-1-j)}\left\{-(E_{km})^T[AG+K^TR]^T\right.\right.$$

$$\left.\left.-[AG+K^TR](E_{km})\right\}\,(F+GK)^{T^{(j)}}\,]\,X_o\right] \qquad (D33)$$

Using the trace properties, we can simplify (D33) as

$$\frac{\partial e}{\partial K_{km}} = -TR\left[\left\{\left(\sum_{i=0}^{n-1}\sum_{j=0}^{n-i-1}(-1)^{i+j}a_i\,((F+GK)^T)^{(j)}\,X_o\,Z^{-1}(F+GK)^{(n-i-1-j)}\right)T\right.\right.$$

$$\left.\left.+\left(\sum_{i=0}^{n-1}\sum_{j=0}^{n-i-1}(-1)^{i+j}a_i\,((F+GK)^T)^{(j)}X_o\,Z^{-1}(F+GK)^{(n-i-1-j)}\right)\right\}\right.$$

$$\left.(AG+K^TR)\,(E_{km})\right] = -Tr[(Y^T+Y)\,(AG+K^TR)\,(E_{km})] \qquad (D34)$$

where

$$Y = \left(\sum_{i=0}^{n-1}\sum_{j=0}^{n-i-1}(-1)^{i+j}a_i\,((F+GK)^T)^{(j)}\,X_o\,Z^{-1}(F+GK)^{(n-i-1-j)}\right)$$

Note that $(Y^T+Y)\,(AG+K^TR)$ is a fixed quantity for a given (F+GK).

The second partial, $\dfrac{\partial^2 e}{\partial K_{ij}\,\partial K_{km}}$ is expressed as

$$\frac{\partial^2 e}{\partial K_{ij}\partial K_{km}} = Tr\left(\frac{\partial^2 A}{\partial K_{ij}\partial K_{km}} X_o\right)$$

$$= -Tr\left[(Y^T+Y)\left(\frac{\partial A}{\partial K_{km}}\right)G(E_{ij})\right]$$

$$-Tr\left[(Y^T+Y)\left(\frac{\partial A}{\partial K_{ij}}\right)G(E_{km})\right]$$

$$-Tr\left[(Y^T+Y)(E_{ij})^T R(E_{km})\right] \qquad (D35)$$

Hence $\dfrac{\partial^2 e}{\partial K_{rs}\partial K_{km}}$

$$= Tr\left[(Y^T+Y)Z^{-1}\left[\sum_{i=0}^{n-1}\sum_{j=0}^{n-i-1}(-1)^{i+j}a_i(F+GK)^{(n-i-1-j)}(E_{km})^T(AG+K^TR)^T(F+GK)^{T^{(j)}}\right]G(E_{rs})\right]$$

$$+ Tr\left[(Y^T+Y)Z^{-1}\left[\sum_{i=0}^{n-1}\sum_{j=0}^{n-i-1}(-1)^{(j)}a_i(F+GK)^{(n-i-1-j)}(AG+K^TR)(E_{km})(F+GK)^{T^{(j)}}\right]G(E_{rs})\right]$$

$$+ Tr\left[(Y^T+Y)Z^{-1}\left[\sum_{i=0}^{n-1}\sum_{j=0}^{n-i-1}(-1)^{i+j}a_i(F+GK)^{(n-i-1-j)}(E_{rs})^T(AG+K^TR)^T(F+GK)^{T^{(j)}}\right]G(E_{km})\right]$$

$$+\mathrm{Tr}\left[(Y^T+Y)Z^{-1}\left[\sum_{i=0}^{n-1}\sum_{j=0}^{n-i-1}(-1)^{i+j}a_i\,(F+GK)^{(n-i-1-j)}\,(AG+\right.\right.$$

$$\left.\left.K^TR)\,(E_{rs})(F+GK)^{T^{(j)}}\right]\,G(E_{km})\right]$$

$$-\mathrm{Tr}\left[(Y^T+Y)\,(E_{rs})^T\,R(E_{km})\right] \qquad \text{(D36)}$$

As we can see from (D36), it is required to compute $(F+GK)^j$ to determine the second partials. If we store all powers of (F+GK), it requires 20 x 400 = 8,000 words memory for a 20th-order system. On the other hand, we can compute $(F+GK)^j$ from (F+GK) stored in the memory which requires only 400 words memory but this requires a large amount of computing time.

REFERENCES

D1. R. Bellman, Introduction to Matrix Analysis, McGraw-Hill Book Company, Inc., New York, 1960, p. 227.

D2. C.F. Chen and L.S. Shieh, "A Note on Expanding $PA+A^TP = -Q$," IEEE Trans. Automatic Control (Correspondence), Vol. AC-13, pp. 122-123, February 1968.

D3. S.P. Bingulac, "An Alternate Approach to Expanding $PA+A^TP = -Q$," IEEE Trans. Automatic Control (Correspondence), Vol. AC-15, pp. 135-137, February 1970.

D4. A. Jameson, "Solution of the Equation AX+XB = C by Inversion of an MxM or NxN Matrix," SIAM J. Appl. Math., Vol. 16, pp. 1020-1023, September 1968.

APPENDIX E
PARAMETER VARIATIONS EXAMPLE

Numerical results obtained in the study of sensitivity to parameter variations for the C-5A example are presented in this appendix. These results demonstrate the applicability of design of optimal insensitive controllers to a realistic problem with significant range of admissible parameters. The limitation in the magnitude of the parameter range is apparent for this problem.

The model of the vehicle used in this study is described, followed by a description of the problem treated. Numerical results indicating the nature of dependence of performance on parameter variations is then summarized. Finally an analysis of eigenvalue dependence on parameter variations is presented which indicates limits on the admissible range of parameter variations for which the proposed approach to insensitivity is valid.

MODEL DESCRIPTION

The longitudinal axis of the C-5A, a large transport aircraft, was chosen for the constant coefficient example described in Section II. This vehicle was chosen because of the inability in a previous study to derive a simplified controller from an optimal controller. Also the data was readily available. The single flight condition treated is the low-altitude cruise condition with 50 percent fuel and 50 percent cargo indicated in Reference 10. The original data available consisted of three rigid-body modes, fifteen symmetric structural flexure modes described in Table E1, lift-growth effects expressed in terms of Wagner and Kussner functions for the wing, horizontal tail, vertical tail and fuselage and gust penetration effects represented by time delays. A design model was derived for the optimization study reported in Reference 10. This model included two rigid-body modes (vertical velocity and pitch rate),

structural bending modes 1, 3, 4, 6, 7 and 11, Kussner lift-growth modes, a second-order wind filter and three first-order actuators. The expected value of a quadratic form in stresses and stress rates at two wing stations and three fuselage stations, including the horizontal tail root, and normal accelerations at four fuselage stations was chosen as the performance index. Aileron, inboard elevator and spoiler control surfaces were used for control.

Table E1. C-5A Mode Descriptions

Mode	Frequency (Hz)	Description
1	0.750	First wing vertical bending
2	1.780	First wing chordwise bending
3	2.328	Second wing vertical bending
4	2.617	First fuselage vertical bending
5	2.814	Outboard pylon lateral bending
6	3.061	Inboard pylon lateral bending
7	3.196	Third wing vertical bending
8	4.119	First wing torsion
9	5.013	Outboard pylon torsion
10	5.155	Inboard pylon torsion
11	5.387	Second fuselage vertical bending
12	6.024	Horizontal stabilizer vertical bending
13	6.702	Second wing chordwise bending
14	6.894	Fourth wing vertical bending
15	7.891	Second wing torsion

The design model was used as the constant coefficient example in Section II. The state variables chosen were vertical velocity, pitch rate, structural mode displacements and rates, control surface displacements, and wind model and Kussner function variables. Specifically, the model used in Section II consisted of the state vector

$$[w, q, \eta_1, \dot{\eta}_1, \eta_3, \dot{\eta}_3, \eta_4, \dot{\eta}_4, \eta_6, \dot{\eta}_6, \eta_7, \dot{\eta}_7, \eta_{11}, \dot{\eta}_{11}, \delta_a, \delta_e, \delta_s, p_1, p_2, p_3, p_4, p_5, \alpha_g]^T$$

the control vector

$$[\dot{\delta}_a, \dot{\delta}_e, \dot{\delta}_s]^T$$

the response vector, r, a 14-vector

$$r = Hx + Dn$$

and the performance index $J = E\{\int_0^\infty r(t)^T Q r(t) dt\}$.

This model was modified to study the effect of parameter variations. To reduce the size of the problem the three highest-frequency flexure modes (6, 7 and 11) were eliminated. Also, the Kussner and wind states (p_1, p_2, p_3 p_4, p_5, α_g) were eliminated, and the deterministic problem was treated. This left an eleven-dimensional state vector

$$x^T = [x_1, x_2, \ldots, x_{11}]^T = [w, q, \eta_1, \dot{\eta}_1, \eta_3, \dot{\eta}_3, \eta_4, \dot{\eta}_4, \delta_a, \delta_e, \delta_s]^T$$

As initial conditions for this problem we chose $x_i(0) = \sqrt{x_{ii}}$ where x_{ii} denotes the corresponding diagonal element of the 23rd-order state covariance matrix corresponding to the optimal control with complete measurement used as an initial point for the simplification of Section II. For the parameter variations analysis the performance index was chosen to be

$$\hat{J} = \int_0^\infty \{[x(t)]^T \hat{Q}\, x(t) + [u(t)]^T R\, u(t)\}\, dt \qquad \text{(E1)}$$

where $\hat{Q} = (\hat{H})^T \hat{Q} \hat{H}$, $R = D^T QD$ and $\hat{H}$ is obtained from the original H matrix by deleting each column corresponding to a deleted state. The cross-product term $2\, x^T (\hat{H})^T QDu$ was omitted for convenience even though $(\hat{H})^T QD$ was nonzero. The parameter variations considered were percentage changes in the undamped natural frequencies associated with the three flexure modes. Nominal values for these frequencies were assumed to be:

ω_1 = frequency of first flexure mode = 5.6 rad/sec

ω_3 = frequency of third flexure mode = 14.5 rad/sec

ω_4 = frequency of fourth flexure mode = 17.0 rad/sec

The vector equation of motion was expressed in the form

$$\dot{x} = Fx + Gu \tag{E2}$$

where the following explicit dependence on the ω_i was assumed:

$$f_{43} = -(\omega_1)^2, \qquad f_{65} = -(\omega_3)^2, \qquad f_{87} = -(\omega_4)^2$$

$$f_{44} = -2\zeta_1\omega_1, \qquad f_{66} = -2\zeta_3\omega_3 \qquad f_{88} = -2\zeta_4\omega_4$$

PROBLEM DESCRIPTION

According to theorems 1 and 2 of Section III the optimal insensitive controller for a small convex region of admissible system parameters is the optimal controller corresponding to a set of parameter values belonging to the boundary of the convex region which yields the maximum optimal cost. The theory gives no quantitative estimate of the possible range of parameters. Similarly, the derivations in Appendix D yield qualitative rather than quantitative results concerning computational time requirements and approximation accuracy. Thus, the major purpose for treating this example was to obtain quantitative data from a realistic problem.

Two specific questions to be answered with this example were:

- Is it computationally feasible to compute a second order approximation to the optimal performance surface?
- For how large a range of admissible parameters is such an approximation adequate to define an optimal insensitive controller?

Using the adjoint method described in Appendix D, the answer to the first question was yes. In answer to the second question the approximation was adequate for ± 15%, but inadequate for ± 20% variations from nominal values.

The optimal cost surface is

$$J^* = \mathrm{Tr}\,\{P x_o x_o^T\} \tag{E3}$$

where P is the matrix function of the parameters which satisfies

$$PF + F^T P - P(GR^{-1}G^T)P + Q = 0 \tag{E4}$$

The second-order approximation to J* is

$$\tilde{J}^* = \mathrm{TR}\,\{P(\omega_1, \omega_2, \omega_3)X_o\} + \nabla_\omega J^*(\Delta\omega) + \frac{1}{2}(\Delta\omega)^T \nabla_{\omega\omega} J^*(\Delta\omega) \tag{E5}$$

where

$$\Delta\omega = \begin{bmatrix} \Delta\omega_1 \\ \Delta\omega_3 \\ \Delta\omega_4 \end{bmatrix} = \text{the vector of flexure mode frequency variations,}$$

ω_1, ω_3, ω_4 are nominal values of the flexure mode frequencies,

$$\nabla_\omega J^* = \left[\frac{\partial J^*}{\partial \omega_1}, \frac{\partial J^*}{\partial \omega_3}, \frac{\partial J^*}{\partial \omega_4}\right] \text{ evaluated at } \omega_1, \omega_3, \omega_4, \text{ and } \nabla_{\omega\omega} J^*$$

is the matrix of second partials of J evaluated at the point ω_1, ω_3, ω_4.

The partials of J* were computed from the following equations (which are derived in Appendix D):

$$\frac{\partial J^*}{\partial \omega_i} = 2\,\mathrm{Tr}\left\{VP\frac{\partial F}{\partial \omega_i}\right\}$$

$$\frac{\partial^2 J^*}{\partial \omega_i \partial \omega_j} = \mathrm{Tr}\left\{\frac{\partial F}{\partial \omega_i} V \frac{\partial P}{\partial \omega_j} + \frac{\partial F}{\partial \omega_j} V \frac{\partial P}{\partial \omega_i}\right\} - \mathrm{Tr}\left\{\frac{\partial P}{\partial \omega_i} GR^{-1}G^T \frac{\partial P}{\partial \omega_j} V\right\}$$

where P satisfies (E-4), V is defined by

$$(F-GR^{-1}G^TP)V + V(F-GR^{-1}G^TP) + X_o = 0$$

and $\frac{\partial P}{\partial \omega_i}$ satisfies

$$(F-GR^{-1}G^TP)^T \frac{\partial P}{\partial \omega_i} + \frac{\partial P}{\partial \omega_i}(F-GR^{-1}G^TP) + P\frac{\partial F}{\partial \omega_i} + \left\{P\frac{\partial F}{\partial \omega_i}\right\}^T = 0$$

for i = 1, 3, 4. Numerical results from these equations are presented below.

NUMERICAL RESULTS

The first and second partials were evaluated to yield the following approximation:

$$\tilde{J}^* = 21204 + [206, -589, -366]\begin{bmatrix}\Delta\omega_1\\ \Delta\omega_3\\ \Delta\omega_4\end{bmatrix} + [\Delta\omega_1, \Delta\omega_3, \Delta\omega_4]\begin{bmatrix}-1.45 & -2.63 & -10.24\\ -2.63 & 10.66 & 15.46\\ -10.24 & 15.46 & 24.88\end{bmatrix}\begin{bmatrix}\Delta\omega_1\\ \Delta\omega_3\\ \Delta\omega_4\end{bmatrix} \quad \text{(E6)}$$

For $\Delta\omega$ in the cube Ω defined by

$$\Omega = \{\Delta\omega : |\Delta\omega_i| \le 0.15\,\omega_i,\ i = 1,3,4\}$$

the first-order term of (E-6) dominates the second-order term. The point in Ω which minimizes $\tilde{J}^*$ is readily determined to be

$$\Delta\omega_1 = 0.15\omega_1,\ \Delta\omega_3 = -0.15\omega_3,\ \Delta\omega_4 = -0.15\omega_4$$

The corresponding $\tilde{J}^* = 24029$ compares quite well with the corresponding value of $J^* = 24813$. Values of J^* and $\tilde{J}^*$ at various points in Ω are listed in Table E2. The accuracy of the estimated value of J^* is reasonably good as is evident from Table E2. Also $\tilde{J}^*$ identifies extremal points of J^* correctly.

Table E2. Optimal Cost and Estimated Cost by Taylor Series Expansion at Nominal F

Parameter Variations from Nominal Values (%)			Optimal Cost, J^*	Estimated Cost by Taylor Series Expansion of Nominal F, $\tilde{J}^*$
$\Delta\omega_1$	$\Delta\omega_3$	$\Delta\omega_4$		
15	-15	-15	24813	24029
15	-15	15	21670	21733
15	15	15	19608	19491
15	15	-15	21089	21108
-15	-15	15	21447	21454
-15	-15	-15	24091	23575
0	0	0	21264	21264
-10	-10	-10	22932	22706
10	10	10	20088	19950
-15	15	-15	20598	20680
-15	15	15	19405	19252

To see whether the optimal controller at the maximum cost J^*_{max} at 15% ω_1, -15% ω_3 and -15% ω_4 from the nominal F was indeed insensitive five points in the convex region of the parameter variation were selected, and their optimal controllers were obtained. The cost J of the controllers for various points in the region were determined. As shown in Table E3, the optimal controller at the maximizing point of J* yields lower costs for the points than J^*_{max}, whereas the optimal controller at J^*_{min} produces three points whose costs exceed J^*_{max}.

An attempt was made to perform the same analysis for an enlarged Ω variation of 20% of nominal values. The resulting optimal cost values and costs for the optimal controller at the maximizing points displayed in Table E4 show that this cube is too large. The optimal controller for the maximizing point yields unstable systems for two vertices of Ω. Also, the cost at one vertex exceeds the maximum value of J*. This indicates that one would have to "back-off" from the maximizing point to derive a satisfactory controller. One computationally feasible way of analyzing this behavior is to analyze the locus of roots as the point defining the optimal controller varies. Such an analysis was performed for the 15% cube.

Comparing Table E3 and Figure E1, we can conclude that along the diagonal line from P5 = (+15% ω_i, -15% ω_3, -15% ω_4) to P1 = (-15% ω_i, +15% ω_3, +15% ω_4) the optimal cost J* increases. The root loci of the optimal controller of P5 for the points along the diagonal line are given in Figure E2. As seen in Figures E2 and E3, if ω_1, ω_3, and ω_4 change further than -15% ω_1, +15% ω_3, +15% ω_4 in the same direction, then the damping ratio decreases and finally one actuator pole goes to the right half plane, and hence the system becomes unstable. At this point a tradeoff between stability and performance would have to be made for the 20% cube. Fortunately, the stability margin need not be large since parameter variations have already been included.

Table E3. Costs of Controllers at Various Points

Parameter Variations from Nominal Values			Optimal Cost, J*	Cost due to the Optimal Controller of				
				$\Delta\omega_1$ 15% ω_1 $\Delta\omega_3$ -15% ω_3 $\Delta\omega_4$ -15% ω_4	10% ω_1 -10% ω_3 -10% ω_4	5% ω_1 -5% ω_3 -5% ω_4	0 % ω_1 0% ω_3 0% ω_4	-15% ω_1 15% ω_3 15% ω_4
$\Delta\omega_1$	$\Delta\omega_3$	$\Delta\omega_4$						
+15%	-15%	-15%	24813	24813	24888	25128	25590	29855
+15%	-15%	+15%	21670	22455	22213	22128	22146	22728
+15%	+15%	+15%	19608	22036	20852	20269	19948	19641
+15%	+15%	-15%	21089	22392	22200	22127	22111	22577
+10%	+10%	+10%	20088	21957	20967	20500	20259	20143
+10%	-10%	-10%	23301	23375	23301	23366	23570	25675
+ 5%	- 5%	- 5%	22162	22478	22226	22162	22219	23267
0%	0%	0%	21264	22072	21528	21319	21265	21743
-10%	-10%	-10%	22932	23118	22994	23066	23284	25649
-15%	-15%	+15%	21447	22454	21921	21767	21778	22399
-15%	-15%	-15%	24091	24364	24398	24655	25163	30799
-15%	+15%	-15%	20598	23049	22309	22303	22438	23268
-15%	+15%	+15%	19405	24730	21613	20374	19822	19405

Note: "□" means that the value exceeds J^*_{max} = 24813.

Table E4. Costs of the Optimal Controller of ($\Delta\omega_1 = 20\%\ \omega_1$, $\Delta\omega_3 = -20\%\ \omega_3$, $\Delta\omega_4 = -20\%\ \omega_4$) at Various Points

Parameter Variations from Nominal Values			Optimal Cost, J*	Cost due to the Optimal Controller of ($\Delta\omega_1 = 20\%\ \omega_1$, $\Delta\omega_3 = -20\%\ \omega_3$, $\Delta\omega_4 = -20\%\ \omega_4$)
$\Delta\omega_1$	$\Delta\omega_3$	$\Delta\omega_4$		
+20%	-20%	-20%	26853	26853
-20%	-20%	-20%	25566	27110
+20%	-20%	+20%	21756	23039
+20%	+20%	+20%	19200	25222
+20%	+20%	-20%	21018	23593
-20%	-20%	+20%	21505	23941
0%	0%	0%	21204	23501
+10%	+10%	+10%	20088	24346
+ 5%	- 5%	- 5%	22172	23146
-20%	+20%	-20%	20300	unstable
-20%	+20%	+20%	18968	unstable

Note: "[ ]" means that the value exceeds $J^*_{max} = 24813$.

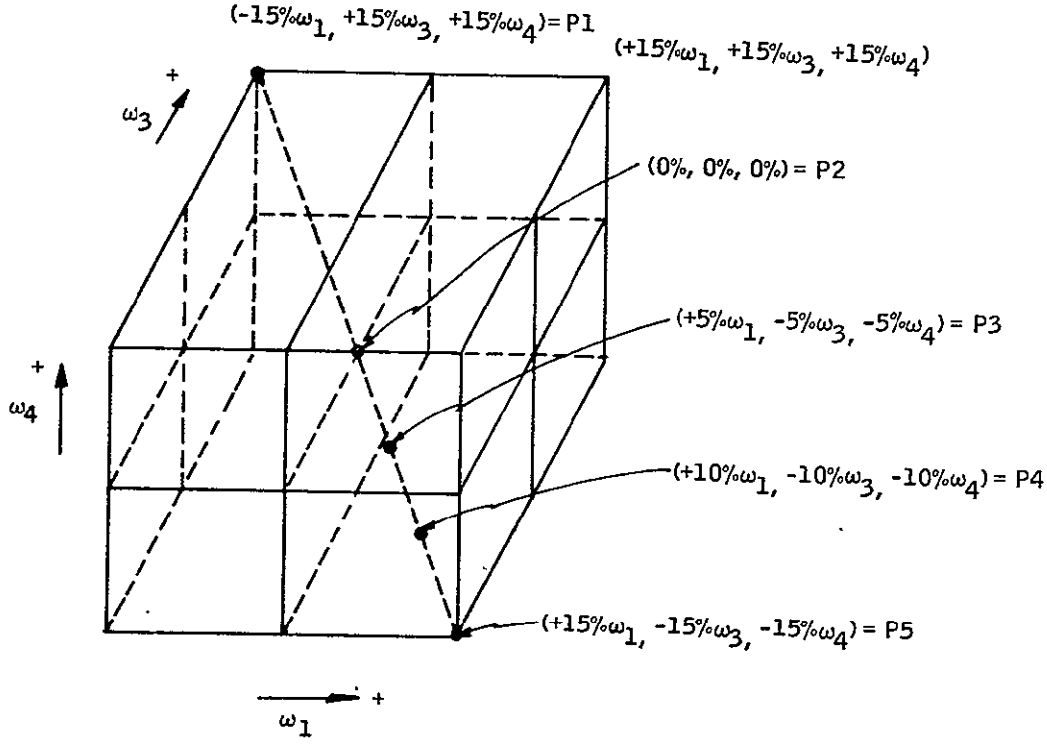


Figure E1. A Convex Range of Parameter Variations

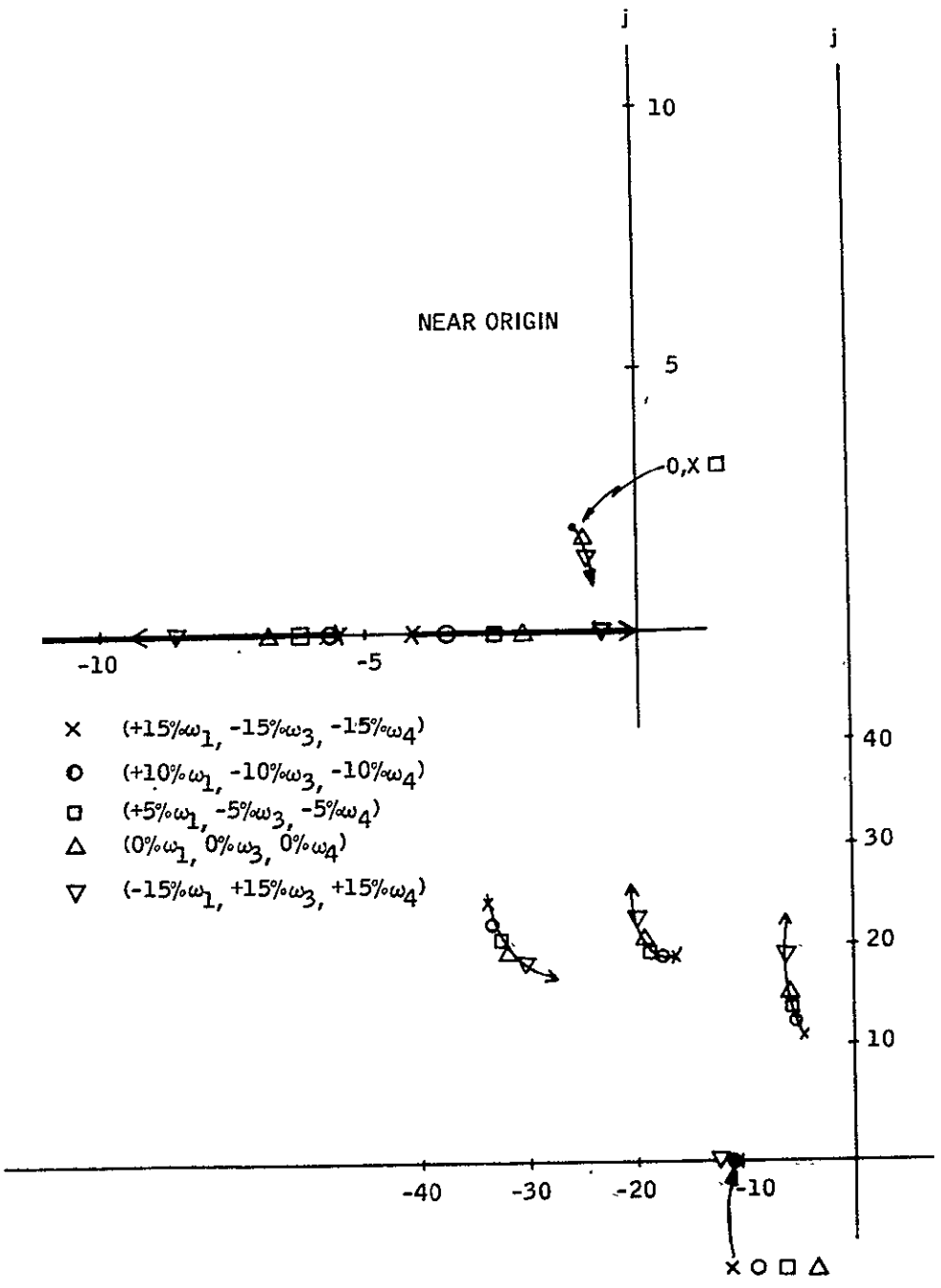


Figure E2. Root Loci of the Optimal Controller of ($\Delta\omega_1 = 15\%\,\omega_1$, $\Delta\omega_3 = -15\%\,\omega_3$, $\Delta\omega_4 = -15\%\,\omega_4$) for Five Points

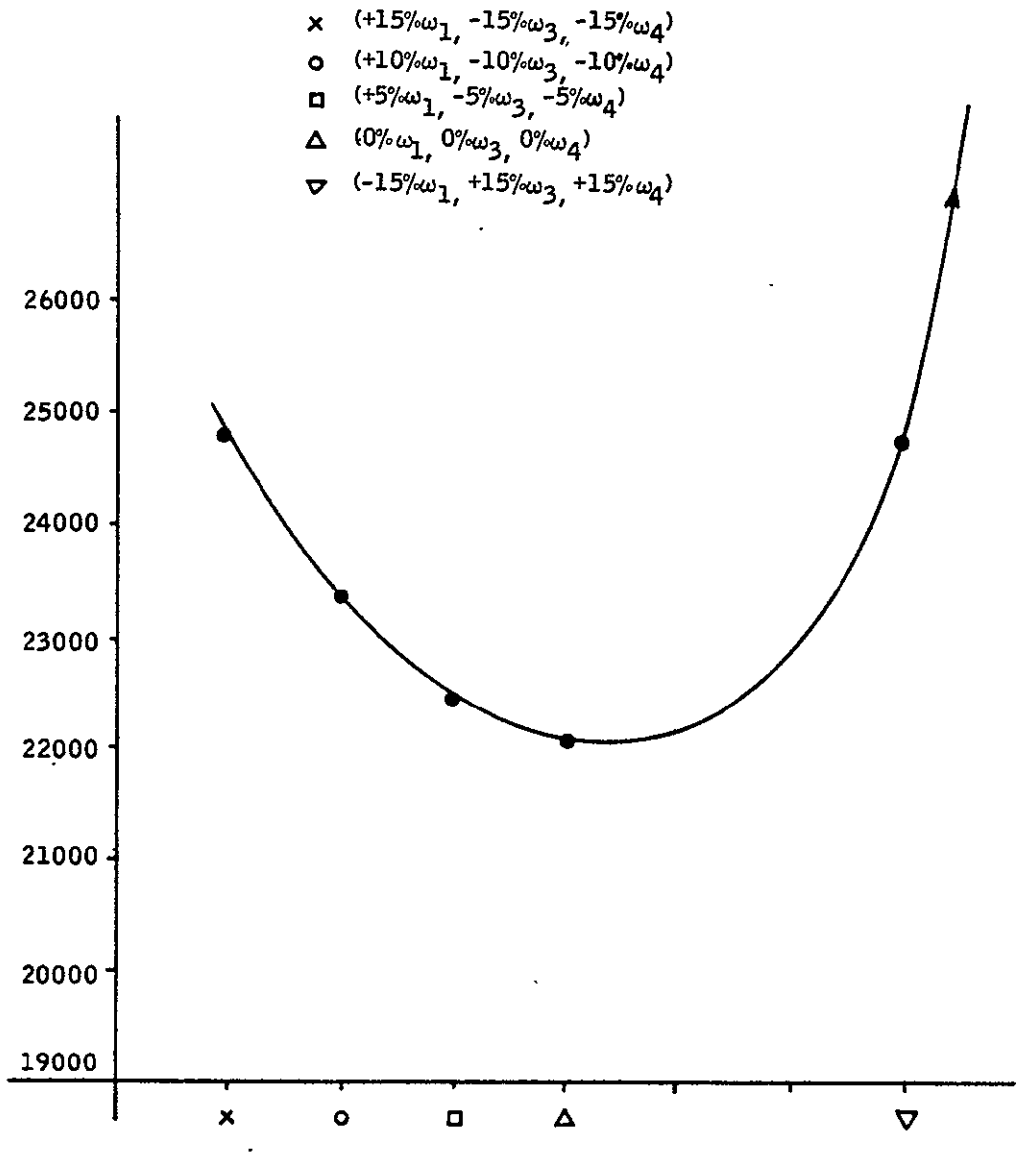


Figure E3. Costs of the Optimal Controller of J^*_{max} for Five Points Along the Diagonal Line Between J^*_{max} Point and J^*_{min} Point

APPENDIX F
DERIVATION OF THE CHARACTERISTIC POLYNOMIAL

This appendix derives the characteristic polynomial for an arbitrary multi-input multi-output feedback control system. We consider plants of the following form:

$$\begin{aligned} \dot{x} &= Fx + Gu \qquad (F,\ G) \text{ controllable, rank } (G) = r \\ z &= Mx \qquad\quad (F,\ M) \text{ observable, rank } (M) = m \end{aligned} \tag{F1}$$

where x is an n-dimensional state vector, u an r-dimensional control vector and z an m-dimensional vector of measured outputs. Without loss of generality, we will assume that the matrix G has a single nonzero entry in each column. This will always be the case in aircraft or launch booster models where each input drives a separate actuator. Moreover, even in cases where G contains additional nonzero entries, we can always construct a similarity transformation T such that

$$\begin{aligned} x &= Tv \\ \dot{v} &= T^{-1} F\, Tv + T^{-1} Gu \\ z &= HTv \end{aligned} \tag{F2}$$

where the new matrix $T^{-1}G$ has the form

$$T^{-1}G = \left[\begin{array}{c} 0 \\ \hline I_{rr} \end{array}\right] \tag{F3}$$

This is possible whenever rank (G) = r. It can be readily verified by letting $T = [T_1 \,|\, G]$, where T_1 is any collection of (n-r) independent columns, independent of G.

Let the r nonzero entries of G be denoted by G_{ij} where j = 1, 2, ..., r, and where i takes on r values depending upon j, i.e., i = i(j) = i(1), i(2), ..., i(r). Then the closed-loop system matrix for a linear fixed-form controller u = Kz will take the following form

$$(F + GKM)^{(i)} = \begin{cases} F^{(i)} & i \neq i(j) \text{ for any } j \\ F^{(i)} + \sum_{\ell} G_{ij} K_{j\ell} M^{(\ell)} & i = i(j) \\ & j = 1, 2, \ldots, r \end{cases} \tag{F4}$$

where the symbol $A^{(i)}$ denotes the i^{th} row of matrix A.

In words, the closed loop system matrix is the original matrix F with linear combinations of rows of M added to its $i(j)^{th}$ rows.

Now consider the characteristic equation of the closed loop system. This is given by

$$\det (sI - F - GKM) = o \tag{F5}$$

In order to deal with this equation, we will introduce the following notation

$$\text{DET } [a^{(1)}, a^{(2)}, \ldots, a^{(r)}] = \det (\Delta)$$

where

$$\Delta^{(i)} = \begin{cases} (sI - F)^{(i)} & i \neq i(j) \\ -a^{(i)} & i = i(j) \text{ for} \\ & j = 1, 2, \ldots, r \end{cases} \tag{F6}$$

Further, we will let $\phi^{(j)} = (sI - F)^{(i(j))}$, $j = 1, 2, \ldots, r$.

Verbally, this notation denotes the determinant of the matrix $(sI - F)$ with rows $i(1), i(2), \ldots, i(r)$ replaced by $-a^{(1)}, -a^{(2)}, \ldots -a^{(r)}$, respectively. Then it follows that

$$\det (sI - F - GKM) = \text{DET} \left[\phi^{(1)} + \sum_{\ell} G_{i(1)1} K_{1\ell} M^{(\ell)}, \ldots, \phi^{(r)} + \sum_{\ell} G_{i(r)r} K_{r\ell} M^{(\ell)}\right] \tag{F7}$$

We will now show that equation (F7) corresponds to at most an r^{th} order polynomial in the gains $K_{j\ell}$. This will be done iteratively, first for $r = 1$, then $r = 2$, and finally for general r.

For $r = 1$, note that we are dealing with the determinant of a matrix whose $i(1)^{th}$ row is a sum of several terms. Using a well known property of determinants (see, for example, Hadley, Linear Algebra, page 93), this gives

$$\text{DET} \left[\phi^{(1)} + \sum_{\ell} G_{i(1)1} K_{1\ell} M^{(\ell)}\right] = \text{DET} \left[\phi^{(1)}\right] + \sum_{\ell} K_{1\ell} \text{DET} \left[G_{i(1)1} M^{(\ell)}\right] \tag{F8}$$

For higher values of r, we simply apply equation (F8) recursively. For example, $r = 2$ yields

$$\text{DET}\left[\phi^{(1)} + \sum_{\ell} G_{i(1)1}\, K_{1\ell}\, M^{(\ell)},\ \phi^{(2)} + \sum_{\ell} G_{i(2)2}\, K_{2\ell}\, M^{(\ell)}\right] =$$

$$= \text{DET}\left[\phi^{(1)},\ \phi^{(2)} + \sum_{\ell} G_{i(2)2}\, K_{2\ell}\, M^{(\ell)}\right] + \sum_{\ell} K_{1\ell}\ \text{DET}\left[G_{i(1)1}\, M^{(\ell)},\ \phi^{(2)} + \sum_{k} G_{i(2)2}\, K_{2k}\, M^{(k)}\right] \qquad \text{(F9)}$$

$$= \text{DET}\left[\phi^{(1)},\ \phi^{(2)}\right] + \sum_{\ell} K_{2\ell}\ \text{DET}\left[\phi^{(1)},\ G_{i(2)2}\, M^{(\ell)}\right]$$

$$+ \sum_{\ell} K_{1\ell} \left\{ \text{DET}\left[G_{i(1)1}\, M^{(\ell)},\ \phi^{(2)}\right] + \sum_{k} K_{2k}\ \text{DET}\left[G_{i(1)1}\, M^{(\ell)},\ G_{i(2)2}\, M^{(k)}\right] \right\}$$

Similarly, for general values of r, we get

$$\text{DET}\left[\phi^{(1)} + \sum_{\ell} G_{i(1)1}\, K_{1\ell}\, M^{(\ell)},\ \ldots,\ \phi^{(r)} + \sum_{\ell} G_{i(r)r}\, K_{r\ell}\, M^{(\ell)}\right] =$$

$$= \text{DET}\left[\phi^{(1)},\ \ldots,\ \phi^{(r)}\right] + \sum_{j=1}^{r} \sum_{\ell=1}^{m} K_{j\ell}\ \text{DET}\left[\phi^{(1)},\ \ldots,\ G_{i(j)j}\, M^{(\ell)},\ \ldots,\ \phi^{(r)}\right]$$

$$+ \underbrace{\sum_{j_1 j_2}} \sum_{\ell_1=1}^{m} \sum_{\ell_2=1}^{m} K_{j_1\ell_1}\, K_{j_2\ell_2}\ \text{DET}\left[\phi^{(1)},\ldots, G_{i(j_1)j_1}\, M^{(\ell_1)},\ldots, G_{i(j_2)j_2}\, M^{(\ell_2)},\ldots,\phi^{(r)}\right]$$

Sum over all nonrepeated pairs of controls

$$\vdots \qquad \text{(F10)}$$

$$+ \underbrace{\sum_{j_1 \cdots j_p}} \sum_{\ell_1=1}^{m} \cdots \sum_{\ell_p=1}^{m} K_{j_1\ell_1} \cdots K_{j_p\ell_p}\ \text{DET}\left[\phi^{(1)},\ldots, G_{i(j_1)j_1}\, M^{(\ell_1)},\ldots, G_{i(j_p)j_p}\, M^{(\ell_p)},\ldots,\phi^{(r)}\right]$$

Sum over all nonrepeated groups of p controls (r! /p! (r-p)! terms)

$$\vdots$$

$$+ \sum_{\ell_1=1}^{m} \cdots \sum_{\ell_r=1}^{m} K_{1\ell_1} \cdots K_{r\ell_r}\ \text{DET}\left[G_{i(1)1}\, M^{(\ell_1)},\ \ldots,\ G_{i(r)r}\, M^{(\ell_r)}\right]$$

Interpretations

The various terms of the polynomial (F10) have the following interpretations:

1) $\mathrm{DET}(\phi^{(1)}, \ldots, \phi^{(r)})$ is the n^{th} order characteristic polynomial, D(s), of the open loop system.

2) $\mathrm{DET}(\phi^{(1)}, \ldots, G_{i(j)j} M^{(\ell)}, \ldots, \phi^{(r)})$ with $G_{i(j)j} M^{(\ell)}$ in the j^{th} position is the $(n-1)^{th}$ order (or lower) numerator polynomial of the (j^{th} input)-to-(ℓ^{th} output) transfer function. This can be verified by computing the transfer function:

$$\frac{y_\ell}{u_j}(s) = -M^{(\ell)}(sI - F)^{-1} \begin{bmatrix} 0 \\ \vdots \\ G_{i(j)j} \\ \vdots \\ 0 \end{bmatrix}$$

$$= \frac{1}{D(s)} \sum_{k=1}^{n} -m_{\ell k} \left[i(j)\, k^{th} \text{ cofactor of } sI - F \right] G_{i(j)j}$$

$$= \frac{1}{D(s)} \mathrm{DET}\left[\phi^{(1)}, \ldots, M^{(\ell)}, \ldots, \phi^{(r)}\right] G_{i(j)j}$$

$$= \frac{1}{D(s)} \mathrm{DET}\left[\phi^{(1)}, \ldots, G_{i(j)j} M^{(\ell)}, \ldots, \phi^{(r)}\right]$$

The numerator polynomials will be denoted by $N_\ell^j(s)$.

3) For higher-order terms of equation (F10) the DET [ ] expressions are so-called "coupling numerators" [18], which appear whenever several feedback loops exist simultaneously. We will denote these polynomials by $N^{j_1 j_2 \cdots j_p}_{\ell_1 \ell_2 \cdots \ell_p}(s)$, i.e.,

$$N^{j_1 \cdots j_p}_{\ell_1 \cdots \ell_p}(s) \triangleq \mathrm{DET}\left[\phi^{(1)}, \ldots, G_{i(j_1)j_1} M^{(\ell_1)}, \ldots, G_{i(j_p)j_p} M^{(\ell_p)}, \ldots, \phi^{(r)}\right] \tag{F11}$$

Simplifications

Equation (F10) can be significantly simplified by using the following two properties.

1) $$N^{j_1 j_2 \cdots j_p}_{\ell_1 \ell_2 \cdots \ell_p}(s) = 0$$

whenever two or more elements of the sequence $\ell_1, \ell_2, \ldots, \ell_p$ are identical. This is because the determinant of a matrix with two proportional rows or columns is zero.

2) $$N^{j_1 j_2 \cdots j_p}_{\ell_1 \ell_2 \cdots \ell_p}(s) = (\pm)\, N^{j_1 j_2 \cdots j_p}_{\ell_1' \ell_2' \cdots \ell_p'}(s)$$

whenever the sequence $\ell_1', \ell_2', \ldots, \ell_p'$ is an arbitrary permutation of the sequence $\ell_1, \ell_2, \ldots, \ell_p$. The correct sign is obtained by noting whether the permutation is even (plus sign) or odd (minus sign). This follows from the fact that an interchange of any two rows or columns of a matrix only changes the sign of its determinant.

Using property (1), we can readily conclude that the maximum order in K of the polynomial (F10) is not equal to the number of controls (r) but equals min (r, m). To show this, just compute a higher order term, i.e., let $r \geq p > m$. Then definition (F11) for

$$N^{j_1 \cdots j_p}_{\ell_1 \cdots \ell_p}$$ requires that we substitute some rows of M into (sI - F)

more than once, so the polynomial must be zero. We can thus eliminate a whole host of terms in equation (F10).

Still more terms can be eliminated by using property (2). Note, in particular, that the p^{th} order summations

$$\sum_{\ell_1=1}^{m} \cdots \sum_{\ell_p=1}^{m}$$

range over all possible combinations of p numbers out of m, including groups with repetitions of numbers and groups which are permutations of other groups. The former will yield vanishing polynomials while the latter's polynomials will differ only in sign from other polynomials. We can thus restrict the summation to nonrepeated groups only, and we can collect all groups which are permutations of each other to form a single multiplier of the polynomial corresponding to, say, the naturally ordered permutation. This gives the following reduced characteristic equation:

$$P(s) = D(s) + \sum_{p=1}^{\min(r,m)} \left\{ \sum_{j_1 \cdots j_p} \sum_{\ell_1 \cdots \ell_p} N^{j_1 \cdots j_p}_{\ell_1 \cdots \ell_p}(s) \left[\sum (\pm) K_{j_1 \ell_1} \cdots K_{j_p \ell_p} \right] \right\} \quad \text{(F12)}$$

sum over all permutations of $\ell_1 \cdots \ell_p$ with appropriate sign

sum over all naturally ordered nonrepeated groups of p measurements out of m

sum over all naturally ordered nonrepeated groups of p controls out of r

APPENDIX G
POLE-PLACEMENT EXAMPLES

This appendix describes two pole-placement problems which were used to debug, exercise, and demonstrate the pole-placement algorithm developed in Section IV. The first problem involves rigid-body controller designs for a sixth-order lateral-axis model of the F4 aircraft (Ref. 6), with rigid-body pole placement taken as the design objective. The second problem is concerned with flexure control of a 17th-order pitch-axis model of the C-5A transport. In this case, the design objective is taken to be placement of rigid body and certain critical flexure poles. These examples serve two functions -- (1) they show that the algorithm works, and (2) they illustrate how it can be utilized in control design and sensor choice problems.

THE F4 LATERAL-AXIS PROBLEM

Disregarding servos, flexure, and other high-frequency phenomena, the lateral axes of the F4 can be modeled by the following equations:

$$\dot{x} = Fx + Gu$$

where

$$x = \begin{bmatrix} p \\ r \\ \beta \\ \phi \\ \delta_R \\ \delta_{AS} \end{bmatrix}$$

- p - roll rate (stability axes)
- r - yaw rate (stability axes)
- β - angle of sideslip
- ϕ - angle of bank
- δ_R - rudder actuator deflection
- δ_{AS} - aileron/spoiler actuator deflection

$$u = \begin{bmatrix} u_R \\ u_{AS} \end{bmatrix} \begin{matrix} - \text{ rudder actuator input} \\ - \text{ aileron/spoiler actuator input} \end{matrix}$$

and where the matrices F and G take the following values for a flight condition at Mach = 0.5, altitude = 5000 feet:

$$F = \begin{bmatrix} -1.7680 & 0.4125 & -14.520 & 0.0000 & 2.031 & 8.9520 \\ -0.0007 & -0.3831 & 6.038 & 0.0000 & -3.398 & -0.3075 \\ 0.0016 & -0.9975 & -0.155 & 0.0586 & 0.028 & -0.0036 \\ 1.0000 & 0.0000 & 0.000 & 0.0000 & 0.000 & 0.0000 \\ 0.0000 & 0.0000 & 0.000 & 0.0000 & -20.000 & 0.0000 \\ 0.0000 & 0.0000 & 0.000 & 0.0000 & 0.000 & -10.0000 \end{bmatrix} \qquad G = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

To represent unity gain actuators, the nonzero values in G (i.e., G_{51} and G_{62}) are equal to 20 and 10, respectively. For our own convenience, however, we will consider these values to be unity and scale down all final gains by a factor of 20 or 10, as required.

The dynamics of the above system are characterized by three principal modes:

(1) The spiral mode, corresponding to a root close to the origin (-0.0156)

(2) The roll subsidence mode, corresponding to a root at -1.85

(3) The dutch-roll mode, corresponding to a complex conjugate pair of roots at $-0.219 \pm j\,2.48$

In addition, there are two actuator modes, corresponding to roots at -20 and -10, respectively.

The "quality" of the lateral-axis dynamics is judged to a large degree by the root locations associated with the three principal modes. In particular, the spiral root should correspond closely to a pure integration, as it does in the free aircraft above. This allows pilots to hold nonzero bank angles and turn rates without stick commands.

The roll subsidence root should be fairly large, since it defines the rate of response of roll rate due to aileron inputs. Values of -3 or -4, for example, are much preferred over the rather sluggish -1.85 value. Finally, the dutch-roll roots should be damped at least by a ratio of 0.25 and should exhibit natural frequencies in the vicinity of 2.5 rad/sec. This assures tolerable sideslip and/or roll-rate oscillations in response to lateral control inputs or disturbances. Note that the free airframe has good dutch-roll frequency characteristics but falls short of the damping requirement.

Given these requirements on pole locations, the lateral axis problem makes an attractive pole placement example. In the paragraphs below, we consider the possibility of satisfying the requirements with several different sensor complements. In each case, the algorithm of Section IV was used to assess pole placement capability and to compute controller gains.

Case 1. Four Measurements

The following measurements are usually available on the aircraft:

$$y = Mx = \begin{bmatrix} p_{se} \\ r_{se} \\ a_y \\ \phi \end{bmatrix}$$

- p_{se} - sensed roll rate (body axis roll rate)
- r_{se} - sensed yaw rate (body axis yaw rate)
- a_y - lateral acceleration at the accelerometer station
- ϕ - bank angle

The measurement matrix corresponding to these signals is given below.

$$M = \begin{bmatrix} 0.9985 & -0.0541 & 0.00 & 0 & 0.00 & 0.000 \\ 0.0541 & 0.9985 & 0.00 & 0 & 0.00 & 0.000 \\ 0.6084 & -2.3640 & -27.33 & 0 & -17.85 & -3.695 \\ 0.0000 & 0.0000 & 0.00 & 1 & 0.00 & 0.000 \end{bmatrix}$$

The various terms of the closed-loop characteristic polynomial [equation (44), main text] for this set of measurements were computed with the algorithm and are summarized in Table G1. This table consists of 15 nonzero coefficient vectors -- one zero-order term, D, eight first-order terms of the form N^{j}_{ℓ}, and six second-order terms of the form $N^{j_1 j_2}_{\ell_1 \ell_2}$. Each vector is identified in Table G1 according to its order and the control sequence $\{j_1, \ldots, j_p\}$ and measurement sequence $\{\ell_1, \ldots, \ell_p\}$ used to generate it. From these sequences, the gain multiplier, $\phi_i(K)$, corresponding to each vector can be reconstructed as follows:

$$\text{General multiplier} = \sum (\pm)\, K_{j_1 \ell_1} K_{j_2 \ell_2} \cdots K_{j_p \ell_p}$$

summed over all permutations of

$$\{\ell_1, \ldots, \ell_p\}$$

$$\text{Specific multiplier } (p=1) = K_{j_1 \ell_1}$$

$$\text{Specific multiplier } (p=2) = (K_{j_1 \ell_1} K_{j_2 \ell_2} - K_{j_1 \ell_2} K_{j_2 \ell_1})$$

In a neighborhood of the initial gain $K_o = 0$, the number of arbitrary poles p_{max} is given by the rank of a matrix formed from all the first order terms of Table G1. This rank turns out to be six, so that all of the system poles can be placed arbitrarily. Consequently, the following closed-loop pole assignments were made:

Table G1. Polynomial Terms for F4, Four Measurements

Order K	Control Sequence $\{j_1 j_2 \ldots j_K\}$ Measurement Sequence $\{\ell_1 \ell_2 \ldots \ell_K\}$ Coefficient Vector $(n_1 n_2 \ldots n_6)^T$					
0						
	•3600E 02	•2329E 04	•1760E 04	•6846E 03	•2762E 03	•3231E 02
1	1 1					
	-•1175E 01	•3706E 03	•4082E 02	-•2174E 02	-•2212E 01	•0000E 00
1	1 2					
	•2149E 02	•2915E 02	•6665E 02	•3923E 02	•3283E 01	•0000E 00
1	1 3					
	-•2921E 02	•3957E 04	•2454E 04	•5325E 03	•2112E 03	•1785E 02
1	1 4					
	•3717E 03	•4436E 02	-•1959E 02	-•2031E 01	•0000E 00	•0000E 00
1	2 1					
	•3143E 01	-•9982E 03	-•1453E 03	-•1839E 03	-•8956E 01	•0000E 00
1	2 2					
	-•5803E 02	-•5607E 02	•4580E 01	-•3179E 01	-•1770E 00	•0000E 00
1	2 3					
	•2565E 03	•8493E 03	•6476E 03	•7529E 02	•7615E 02	•3695E 01
1	2 4					
	-•9997E 03	-•1449E 03	-•1838E 03	-•8952E 01	•0000E 00	•0000E 00
2	1 2 1 2					
	•6323E-10	•3351E 01	•2979E 02	•0000E 00	•0000E 00	•0000E 00
2	1 2 1 3					
	-•5819E 01	•1834E 04	•2433E 02	•1517E 03	•0000E 00	•0000E 00
2	1 2 1 4					
	•1812E 00	•1611E 01	•0000E 00	•0000E 00	•0000E 00	•0000E 00
2	1 2 2 3					
	•1075E 03	•9931E 02	-•4928E 01	•1529E 02	•0000E 00	•0000E 00
2	1 2 2 4					
	-•3346E 01	-•2975E 02	•0000E 00	•0000E 00	•0000E 00	•0000E 00
2	1 2 3 4					
	-•1836E 04	-•2403E 02	-•1523E 03	•0000E 00	•0000E 00	•0000E 00

Spiral	$(s + 0)$
Roll subsidence	$(s + 4.0)$
Dutch roll	$(s^2 + 1.25\ s + 6.25)$
Actuator	$(s^2 + 30\ s + 450)$

and the algorithms executed seven Newton-Raphson iterations before the convergence test was satisfied. The final gains were

$$K = \begin{bmatrix} -41.03 & 110.60 & 0.1649 & -28.64 \\ 17.24 & -46.65 & 0.0000 & 0.00 \end{bmatrix}$$

The convergence test required that the spiral root be located to an accuracy of $\pm$ 0.001, the roll subsidence root to an accuracy of $\pm$ 0.01, dutch-roll coefficients to $\pm$ 0.01 and the actuator coefficients to $\pm$ 1.0. An independent check of the closed-loop roots showed that all conditions were satisfied.

It is important to recognize that each iteration of the algorithm requires a square [in this case (6x6)] Jacobian matrix. Such a matrix was obtained from our original (6x8) Jacobian by choosing the first n independent columns and deleting the rest. This arbitrarily removes two degrees of freedom and accounts for the fact that the final values of K_{23} and K_{24} are zero.

Case 2. Three Measurements

Having shown that four measurements are sufficient for arbitrary pole placement, we next proceeded to delete bank angle from the measurement complement. The program was again used to compute coefficient vectors, to assess rank, and to carry out Newton-Raphson iterations.

A bit of reflection will verify that the set of coefficients vector for three measurements must consist of all vectors in Table G1 which do not involve the fourth measurement. This gives ten vectors -- one zero-order term, six first-order terms and three second-order ones. The rank of the first-order vectors is again six, so all system poles can again be assigned arbitrarily. For the same assignments used above, the algorithm proceeded to execute six iterations before terminating with the following gains:

$$K = \begin{bmatrix} 2.39 & -10.44 & -1.19 \\ -16.67 & 83.57 & 6.56 \end{bmatrix}$$

The desired root accuracies were again verified by an independent check of the closed-loop roots.

Case 3. Two Measurements

Knowing that three measurements is enough, one is tempted to try two -- so we deleted the lateral acceleration signal as well as bank angle. This leaves only two rate gyros as the sensor complement.

The set of coefficient vectors is again a subset of Table G1, found by removing all vectors involving measurements 3 or 4. This leaves six vectors -- one zero-order, four first-order, and one second-order. The rank of the first-order terms is now four, so that only partial pole placement is possible. The initial gain $K_o = 0$ was again selected and the spiral, roll subsidence, and dutch-roll roots were defined as above. This exhausts the four roots which may be arbitrarily specified. We simply have to accept whatever the algorithm gives us for the remaining roots. Final gains and unspecified roots were found in three iterations:

$$K = \begin{bmatrix} -0.43 & 3.57 \\ -1.55 & 1.88 \end{bmatrix}$$

$$\delta(s) = s^2 + 27.06\ s + 149.1$$

actuator roots at -19.35, -7.71

We see that even though the actuator poles could not be placed arbitrarily, they take on quite reasonable values.

It appears, therefore, that only two measurements can suffice to achieve modal control of the lateral axes.

THE C-5A FLEXURE PROBLEM

Appendix E contains descriptions of a 23rd-order flexure model for the C-5A pitch axis. This model was utilized here for high-order pole placement studies. However, since gust dynamics and Kussner lift growth states do not enter the feedback loop, these six degrees of freedom were deleted. This leaves a 17th-order model, comprised of two rigid body states, two states each for the 1st, 3rd, 4th, 6th, 7th and 11th flexure modes, and one state each for symmetric aileron, elevator, and spoiler actuators.

Open-loop poles for the 17th-order model are given in Table G2 together with a set of "desirable locations" abstracted from quadratic optimal flexure control designs and from the LAMS controller (Ref. 10). These locations do not correspond to any one controller but were chosen to reflect general trends exhibited by several designs -- namely, that flexure controllers move rigid-body poles, increase damping of modes 1, 4, and 7, and leave the remaining flexure poles pretty well alone. This same trend was taken to be the design objective for the pole-placement computations reported here.

Table G2. Open-Loop and Desirable C-5A Roots

Mode	Open-Loop Roots	Desirable Roots
Rigid Body	$-1.03 \pm j\ 1.31$	$-1.07 \pm j\ 1.80$
1st Flexure	$-0.901 \pm j\ 5.57$	$-4.00 \pm j\ 6.00$
3rd Flexure	$-0.462 \pm j\ 14.5$	Unchanged
4th Flexure	$-1.05 \pm j\ 17.0$	$-4.50 \pm j\ 18.0$
6th Flexure	$-0.430 \pm j\ 19.3$	Unchanged
7th Flexure	$-0.960 \pm j\ 20.3$	$-11.0 \pm j\ 22.0$
11th Flexure	$-4.29 \pm j\ 35.7$	Unchanged

A sensor complement composed of two accelerometers, one rate gyro, and one surface position transducer was used in the computations. The accelerometers were located on the wing, one near midspan and the other near the wingtip, while the rate gyro was located near the tail. These locations correspond to the LAMS controller (Ref. 10). The position transducer measured aileron actuator deflection. The measured sitnals are therefore summarized by

$$z = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix}$$

- z_1 - midspan acceleration
- z_2 - wing tip acceleration
- z_3 - pitch rate at tail station*
- z_4 - aileron actuator deflection

Each signal is a linear combination of rigid body states, bending mode states, and actuator states.

*Due to an error in programming this signal consisted of rigid-body pitch rate plus mode slopes times mode positions rather than mode rates times mode slopes.

In a neighborhood of the open-loop system, these measurements yield

$$p_{max}\Big|_{K_o=0} = \text{rank}\left[N_1^1 \cdots N_4^1 N_1^2 \cdots N_4^3 \right] \cong 8$$

This is indicated as an approximate value because the numerical determination of the rank of high order matrices is subject to uncertainty. In the present case, the matrix of numerator coefficients spanned 8-dimensional space with certainty but possibly also 9-dimensional space. Components in the 9th coordinate direction, however, were near the least-bit level of the machine and thus of little practical utility. At any rate, eight closed-loop poles could be assigned arbitrarily in a neighborhood of the open-loop system.

Given this information, three separate runs were made with the initial condition $K_o = 0$. The first run placed four poles (rigid body plus 1st flexure), the second six poles (rigid body plus 1st and 4th flexure), and the third placed eight poles (rigid body plus 1st, 4th and 7th flexure). Results are summarized in Table G3. In each case, open-loop actuators at -100 (elevator), -10 (spoiler), and -6 (aileron) were assumed, and a square Jacobian matrix was obtained by allowing only selected feedback gains to vary. The resulting gain structure as well as the number of Newton-Raphson iterations are also shown in Table G3.

As the table indicates, each run achieved its objective of moving a specified group of poles into the desirable locations given in Table G2. However, some of the remaining unassigned poles were destabilized in the process. In Run 1 the damping of 3rd, 4th and 7th flexure modes was reduced; in Run 2 the 7th mode was actually driven unstable; and in Run 3 an actuator root was driven unstable. Fortunately, none of these runs fully exhaust the pole placement capacity offered by the sensors. Away from the origin, the previously weak 9th order of rank strengthened sufficiently to yield

$$p_{max}\Big|_{K_o=K_{run\ 3}} = \text{rank}\left[\sum_i N_i \frac{\partial \phi_i}{\partial K^T} \right] \cong 9,$$

Table G3. Pole Placement Runs for C-5A

Mode	Run 1 4 Poles Placed	Run 2 6 Poles Placed	Run 3 8 Poles Placed	Run 4 9 Poles Placed
Rigid Body	-1.07 ± j 1.80	-1.07 ± j 1.80	-1.07 ± j 1.80	-1.06 ± j 1.79
1st Flexure	-3.95 ± j 6.00	-4.00 ± j 6.03	-4.09 ± j 6.07	-4.00 ± j 5.94
3rd Flexure	-0.253 ± j 8.68	-1.07 ± j 14.9	-0.955 ± j 16.4	-0.506 ± j 15.8
4th Flexure	-0.182 ± j 16.5	-4.49 ± j 18.3	-4.59 ± j 18.0	-4.53 ± j 18.3
6th Flexure	-0.258 ± j 19.3	-0.448 ± j 19.4	-0.357 ± j 19.3	-0.265 ± j 19.3
7th Flexure	-1.59 ± j 18.6	+0.004 ± j 17.8	-10.9 ± j 22.2	-10.9 ± j 22.0
11th Flexure	-4.27 ± j 35.7	-4.54 ± j 36.8	-4.08 ± j 35.0	-4.15 ± j 35.1
Actuators	-99.8 -16.3 - 6.34	-107.0 -2.27 ± j 4.29	-103.0 + 6.15 - 10.1	-108.0 - 0.972 - 2.52
Number of Iterations	11[a]	5	14	20[a]
Gain Structure	Aileron / Elevator / Spoiler z_1 z_2 z_3 z_4 x x x x x	x x x x x x x	x x x x x x x x x	x x x x x x x x x

[a]Total number of iterations in 20-step incremental stepping procedure

which means that one additional pole could be placed arbitrarily. This extra degree of freedom was used in Run 4 to move the unstable actuator pole back into the left-hand plane. The resulting final pole constellation (Table G3) shows that our design objectives can be satisfied by utilizing the full capacity of the sensor complement. All desired locations are achieved and the remaining poles are approximately unchanged. Gain magnitudes associated with Run 4 compare favorably with the LAMS controller.

An interesting byproduct of these pole-placement computations is what might be called a generalized root locus plot. Namely, if an incremental stepping procedure is used to move from the open-loop poles to the desired closed-loop poles, i.e.

$$\lambda(s,\alpha) = \alpha\left[\lambda_{\text{desired}}(s) - \lambda_{\substack{\text{open}\\\text{loop}}}(s)\right] + \lambda_{\substack{\text{open}\\\text{loop}}}(s)$$

$$\alpha = 0,\ \Delta\alpha,\ 2\Delta\alpha,\ \ldots,\ 1$$

then pole positions as a function of α form a familiar single-parameter root locus which shows how unspecified roots migrate as the assigned roots move toward their specified positions. Such plots are shown in Figures G1 and G2 for Runs 1 and 2 of Table G3. Both figures indicate that acceptable levels of damping can be obtained for the unspecified roots if a value of α somewhat less than unity is accepted for the assigned roots.

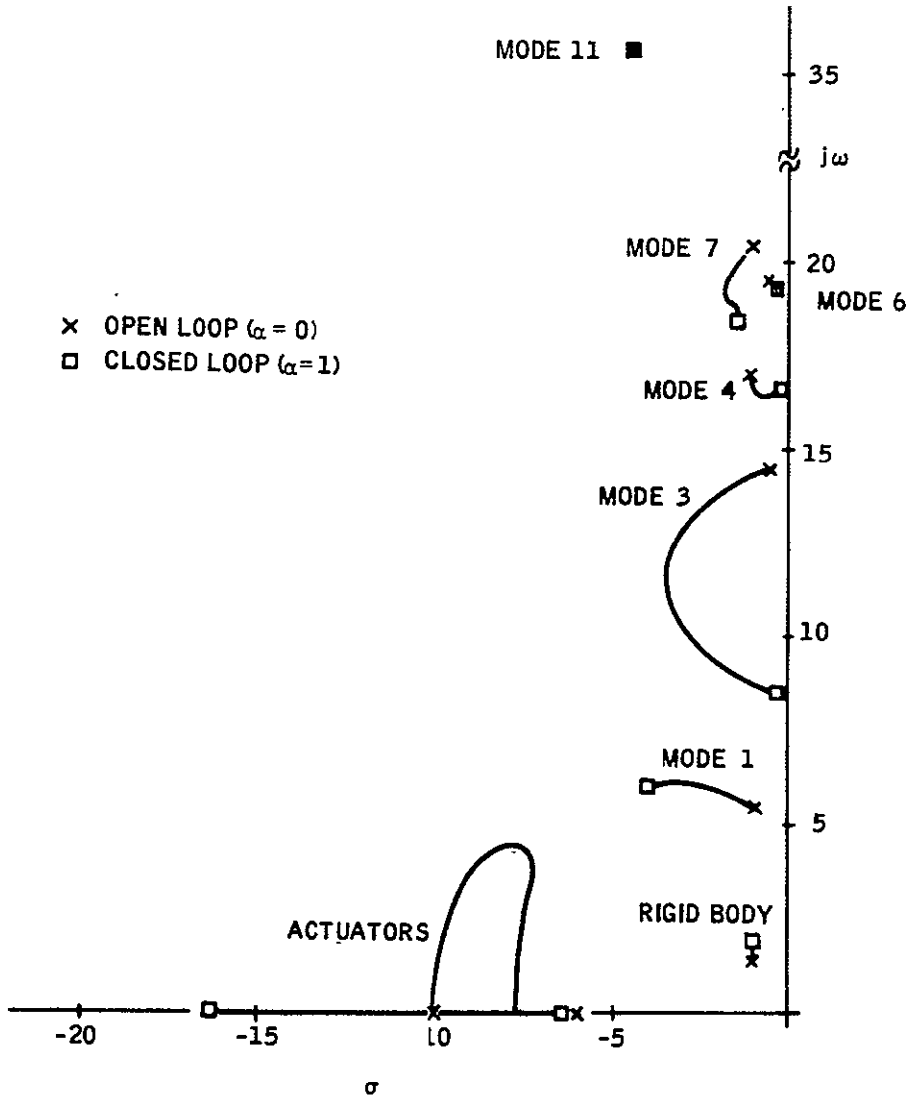


Figure G1. Root Locus: Run 1

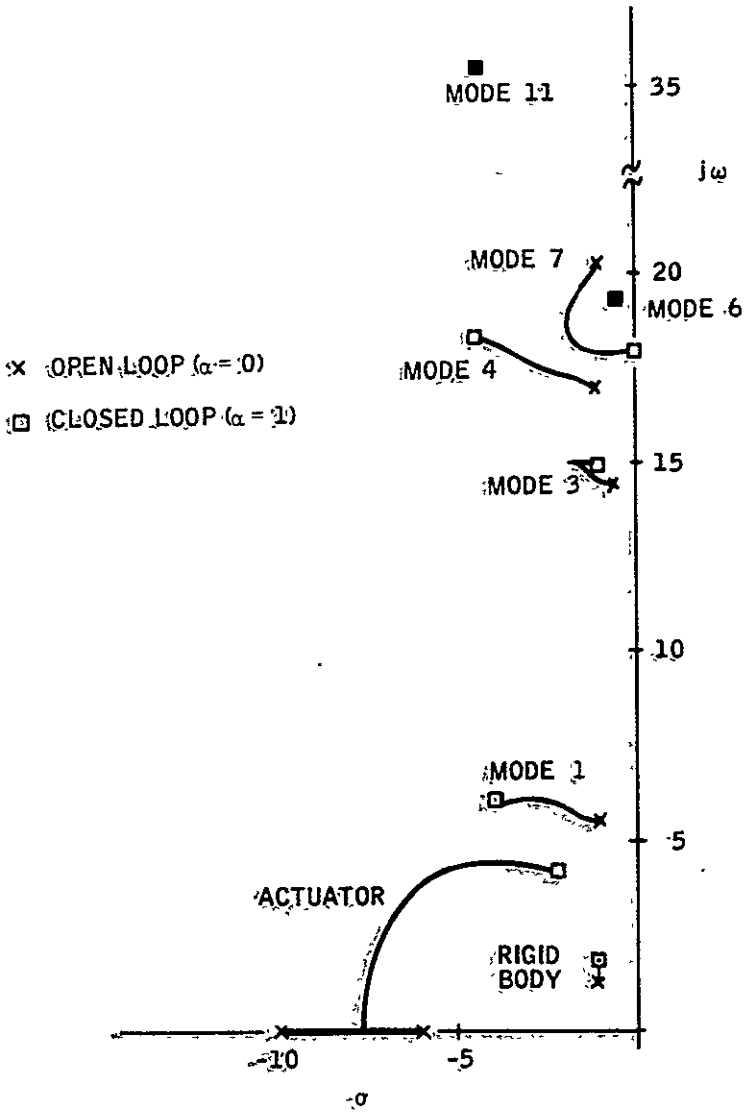


Figure G2. Root Locus: Run 2

APPENDIX H
PROCEDURES FOR COMPUTING COEFFICIENT VECTORS

This appendix describes two methods for computing coefficients of polynomials appearing in the general characteristic equation of Appendix F. These particular computations comprise about one half of the entire computational load associated with the pole placement algorithm of Section IV.

METHOD 1

This procedure is presently used in the algorithm. It mechanizes the computational method of Reference 19 for solving the general eigenproblem det (sB+A), where A and B are arbitrary matrices, possibly singular. (The DET [...] expressions of Appendix F are in this form.) The method consists of reducing the original problem to a lower-order nonsingular one by repeated Gauss reductions of matrices B and A. The nonsingular problem can then be solved by standard methods. The reduction goes like this:

- Via Gaussian elimination, reduce B to

$$B' = \begin{bmatrix} B'_{11} & B'_{12} \\ 0 & 0 \end{bmatrix}$$

where B'_{11} is upper triangular.

All elementary operations performed on B are also performed on A, which gives

$$A' = \begin{bmatrix} A'_{11} & A'_{12} \\ A'_{21} & A'_{22} \end{bmatrix}$$

- Via Gaussian elimination, reduce A' to

$$A'' = \left[\begin{array}{c|c} A''_{11} & A''_{12} \\ \hline 0 & A''_{22} \end{array}\right]$$

where A''_{22} is upper triangular. All elementary operations performed on A' are also performed on B', giving

$$B'' = \left[\begin{array}{c|c} B''_{11} & B''_{12} \\ \hline 0 & 0 \end{array}\right]$$

We now have

$$\det(sB+A) = \pm \det \left\{ s \begin{bmatrix} B''_{11} & B''_{12} \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} A''_{11} & A''_{12} \\ 0 & A''_{22} \end{bmatrix} \right\}$$

$$= \pm \det A''_{22} \det(sB''_{11} + A''_{11})$$

where the sign is determined by row and column operations performed in the two Gaussian elimination steps.

If B''_{11} is nonsingular, the problem has been reduced to a standard one. Otherwise, we let $B = B''_{11}$, $A = A''_{11}$ and return to Step (1).

The remaining nonsingular eigenproblem is presently being solved by computing eigenvalues via QR transformations and reconstructing coefficients from these. Total SDS 9300 computing times are 3 to 5 seconds per 17th-order coefficient vector.

METHOD 2

This method utilizes the fact that higher-order generalized numerators can be expressed in terms of first-order ones. This offers significant computational savings when repeated computations must be performed for different measurement matrices, $M(\Omega, y)$.

Expressions for Higher Order Numerators

An alternate way to derive the characteristic equation of Appendix F is via the transfer function matrix -H(s) relating measurements z and control inputs u; i.e.,

$$z(s) = [-H(s)]u(s)$$

$$u(s) = Kz(s) + u_c(s)$$

$$[1 + H(s)K]z(s) = [-H(s)]u_c(s)$$

$$P(s) = D(s) \det (1 + H(s)K)$$

Expanding this polynomial and comparing the result term for term with equation (F12) of Appendix F provides expressions for the higher order numerators. This process is carried out for a 3-output, r-input system below, from which the general relationships can be deduced.

$$P(s) = D(s) \det \begin{bmatrix} 1 + \Sigma H_{1i} K_{i1} & \Sigma H_{1i} K_{i2} & \Sigma H_{1i} K_{i3} \\ \Sigma H_{2i} K_{i1} & 1 + \Sigma H_{2i} K_{i2} & \Sigma H_{2i} K_{i3} \\ \Sigma H_{3i} K_{i1} & \Sigma H_{3i} K_{i2} & 1 + \Sigma H_{3i} K_{i3} \end{bmatrix}$$

$$= D(s) \left[1 + \Sigma_{11} + \Sigma_{22} + \Sigma_{33} + \right.$$

$$+ \Sigma_{11}\Sigma_{22} - \Sigma_{12}\Sigma_{21} + \Sigma_{11}\Sigma_{33} - \Sigma_{13}\Sigma_{31} + \Sigma_{22}\Sigma_{33} - \Sigma_{23}\Sigma_{32}$$

$$+ \Sigma_{11}\Sigma_{22}\Sigma_{33} - \Sigma_{11}\Sigma_{23}\Sigma_{32} - \Sigma_{12}\Sigma_{21}\Sigma_{33} + \Sigma_{12}\Sigma_{23}\Sigma_{31}$$

$$\left. + \Sigma_{13}\Sigma_{21}\Sigma_{32} - \Sigma_{13}\Sigma_{22}\Sigma_{31} \right]$$

where $\Sigma_{\ell m} = \sum_{i=1}^{r} H_{\ell i} K_{im}$.

Note that the terms which are first-order in Σ may be written as

$$\Sigma_{11} + \Sigma_{22} + \Sigma_{33} = \sum_{k=1}^{3} \sum_{i=1}^{r} H_{ki} K_{ik} = \sum_{j_1} \sum_{\ell_1} H_{\ell_1 j_1} K_{j_1 \ell_1}$$

Similarly the second-order terms may be expressed as:

$$\sum_{j_1} \sum_{j_2} H_{1j_1} H_{2j_2} (K_{j_1 1} K_{j_2 2} - K_{j_1 2} K_{j_2 1}) +$$

$$H_{1j_1} H_{3j_2} (K_{j_1 1} K_{j_2 3} - K_{j_1 3} K_{j_2 1}) +$$

$$H_{2j_1} H_{3j_2} (K_{j_1 2} K_{j_2 3} - K_{j_1 3} K_{j_2 2})$$

This expression can be simplified by observing that terms are skew-symmetric in j_1, j_2, i.e., interchanging the roles of j_1 and j_2 in a term merely changes the sign of the term and in particular if $j_1 = j_2$ then the term is zero. Thus the second-order terms are given by:

$$\sum_{j_1} \sum_{j_2>j_1} (H_{1j_1}H_{2j_2} - H_{1j_2}H_{2j_1})(K_{j_1 1}K_{j_2 2} - K_{j_1 2}K_{j_2 1}) +$$

$$(H_{1j_1}H_{3j_2} - H_{1j_2}H_{3j_1})(K_{j_1 1}K_{j_2 3} - K_{j_1 3}K_{j_2 1}) +$$

$$(H_{2j_1}H_{3j_2} - H_{2j_2}H_{3j_1})(K_{j_1 2}K_{j_2 3} - K_{j_1 3}K_{j_2 2})$$

or, in general form

$$\sum_{j_1 j_2} \sum_{\ell_1 \ell_2} \left[\sum_{P_j} (\pm) H_{\ell_1 j_1} H_{\ell_2 j_2}\right] \left[\sum_{P_\ell} (\pm) K_{j_1 \ell_1} K_{j_2 \ell_2}\right]$$

- $\sum_{P_\ell}$: all permutations of $\ell_1 \ell_2$
- $\sum_{P_j}$: all permutations of $j_1 j_2$
- $\sum_{\ell_1 \ell_2}$: all ordered nonrepeated groups of 2 measurements out of 3
- $\sum_{j_1 j_2}$: all ordered nonrepeated groups of 2 controls out of r

Similar expansion and reduction of the third-order terms yields

$$\sum_{j_1 j_2 j_3} \sum_{\ell_1 \ell_2 \ell_3} \left[\sum_{P_j} (\pm) H_{\ell_1 j_1} H_{\ell_2 j_2} H_{\ell_3 j_3}\right] \left[\sum_{P_\ell} (\pm) K_{j_1 \ell_1} K_{j_2 \ell_2} K_{j_3 \ell_3}\right]$$

From these expressions, it follows that the general closed-loop polynomial for r inputs and m outputs is given by

$$P(s) = D(s)\left\{1 + \sum_{k=1}^{\min(r,m)} \sum_{j_1 \cdots j_k} \sum_{\ell_1 \cdots \ell_k} \left[\sum_{P_j} (\pm) H_{\ell_1 j_1} \cdots H_{\ell_k j_k}\right] \left[\sum_{P_\ell} (\pm) K_{j_1 \ell_1} \cdots K_{j_k \ell_k}\right]\right\}$$

and comparing with equation (F12) of Appendix F yields

$$N^{j_1 \cdots j_k}_{\ell_1 \cdots \ell_k}(s) = D(s)\left[\sum_{P_j} (\pm) H_{\ell_1 j_1} H_{\ell_2 j_2} \cdots H_{\ell_k j_k}\right]$$

$$= \frac{1}{D(s)^{k-1}}\left[\sum_{P_j} (\pm)\, N^{j_1}_{\ell_1} N^{j_2}_{\ell_2} \cdots N^{j_k}_{\ell_k}\right]$$

This expression defines all higher-order numerators in terms of first-order ones. Note, however, that the definition involves a sum of $(nk)^{th}$ order polynomials divided by the $n(k-1)^{th}$ order polynomial D^{k-1}. We know from Appendix F that each higher-order numerator is, in fact, an n^{th}-order polynomial (or lower). This means that D^{k-1} must be a perfect factor of the sum of $(nk)^{th}$ order polynomials – a fact which could lead to numerical problems in computing $N^{j_1 \cdots j_k}_{\ell_1 \cdots \ell_k}$ from the first-order terms. The following procedure is suggested:

- Multiply the definition by D^{k-1} and express the result in coefficient-vector-form:

$$D\, N^{j_1 \cdots j_k}_{\ell_1 \cdots \ell_k} = N$$

where $\bar{D}$ is an (nk x n)-dimensional matrix constructed from the coefficients of $D(s)^{k-1}$ and N is the coefficient vector of

$$\sum_{P_j} (\pm) N_{\ell_1}^{j_1} \cdots N_{\ell_k}^{j_k}$$

- Solve for $N_{\ell_1 \cdots \ell_k}^{j_1 \cdots j_k}$ via pseudo inversion, i.e.,

$$N_{\ell_1 \cdots \ell_k}^{j_1 \cdots j_k} = (\bar{D}^T \bar{D})^{-1} \bar{D}^T N$$

This should minimize numerical problems associated with the polynomial division. Note that the quantity $(\bar{D}^T \bar{D})^{-1} \bar{D}^T$ needs to be computed only once. It can be stored and reused for computation of all k^{th}-order numerators for any given set of first-order numerators.

Computation of First-Order Numerators

All that remains now is to compute the first-order numerators. This can be done via Leverrier's algorithm (also called Fadeeva's method†) which computes the determinant and all of the cofactors of $(sI-F)^{-1}$. That is,

†Zadeh, L.A., and C.A. Desoer, Linear System Theory, McGraw-Hill, New York, 1963.

$$(sI-F)^{-1} = \frac{B(s)}{D(s)} = \frac{B_n s^{n-1} + B_{n-1} s^{n-2} + \ldots + B_1}{s^n + d_n s^n + d_{n-1} s^{n-2} + \ldots + d_1}$$

$$B_n = I \qquad d_n = -\text{Trace}\,(B_n F)$$

$$B_{n-1} = B_n F + d_n I \qquad d_{n-1} = -1/2\,\text{Trace}\,(B_{n-1} F)$$

$$\vdots \qquad\qquad \vdots$$

$$B_k = B_{k+1} F + d_{k+1} I \qquad d_k = -\frac{1}{n+1-k}\,\text{Trace}(B_k F)$$

$$\vdots$$

$$B_1 = B_2 F + d_2 I \qquad d_1 = -\frac{1}{n}\,\text{Trace}(B_1 F)$$

$$0 = B_1 F + d_1 I$$

Once all the cofactors are known, individual first-order numerators are obtained from the relation

$$N_\ell^j(s) = \sum_k -M_{\ell k}\, B_{ki(j)}(s)\, G_{i(j)j}$$

This expression shows that repeated runs with different M matrices are very economical, since the B(s) matrix (and also the $(\bar{D}^T\bar{D})^{-1}\bar{D}^T$ expressions above) need not be recomputed for a new M matrix.